mySites.guru

PageBuilder CK File Upload RCE - June 2026

PageBuilder CK below 3.6.0 lets an attacker upload a file to any folder on your Joomla site with no login and run it. The fix is out. Update to 3.6.0 now.

Phil E. Taylor | 27 June 2026

JCE Profiles Hack · CVE-2026-48907 (27th June): Things have cooled down, but we are still seeing waves of attacks and breaches on Joomla sites. Check your site today with mySites.guru.

Read the alert >

PageBuilder CK is one of the more popular free page builders for Joomla, a drag-and-drop layout tool that thousands of sites lean on for their landing pages and content. On 27 June 2026 its developer, Cedric Keiflin, shipped version 3.6.0 with a changelog that says, in full, “IMPORTANT : Fix security issue”. No detail, no fanfare. That terse line is hiding a serious one.

In every version up to and including 3.5.10, PageBuilder CK has an unauthenticated file upload flaw that leads to remote code execution. An attacker with no login can upload a file to your Joomla site, choose which folder it goes in, and then run it. That is full control of the site: stealing data, defacing pages, planting backdoors, or using your server to attack other people. We confirmed it by comparing the old and new code and reproducing the whole chain on a clean test install.

The fix is PageBuilder CK 3.6.0. If any of your Joomla sites run this component, update them now, then read on for what the flaw does and how to check whether you were already hit.

Breaking: this one is hours old. The fix shipped on 27 June 2026 and we are publishing as the patch lands, so the detail here is moving fast. We have not yet recorded this being exploited in the wild. The moment we do, we will add a dedicated check to mySites.guru to find sites hit by it, the same way we did for the JCE profiles hack. Until then, the suspect content tool already has your back, and every affected site is already flagged on the new Important tab (see below).

TL;DR

- **Unauthenticated arbitrary file upload to remote code execution** in PageBuilder CK, a free Joomla page builder. No login required
- Worse than a typical upload bug: the attacker **chooses the destination folder**, so a planted file can land anywhere on the site, not just an images directory
- Affects **every version up to and including 3.5.10**. Fixed in **3.6.0** for current Joomla, released **27 June 2026**. The vendor also back-patched the older lines: **3.1.1** for Joomla 3 and **3.4.10** for Joomla 4, so match the patched version to the Joomla version your site runs
- The only thing in the attacker's way is a **CSRF token**, which any visitor can read straight off the site's own pages. That is not a security barrier
- **No CVE has been assigned yet.** This is a vendor-disclosed advisory. The vendor changelog reads only "IMPORTANT : Fix security issue"
- We confirmed it by **source-diffing 3.5.10 against 3.6.0** and reproducing the upload-to-execution chain on a clean Joomla 5 install. We are not publishing a working exploit
- **Update to 3.6.0 on every affected site, then check for compromise.** Unpublishing the component does not protect you

The fix is out, but the longer a site stays on an old version, the more time automated scanners have to find it. Bots work through lists of Joomla sites looking for exactly this kind of flaw, and your site does not need to be a target to get caught. Update to 3.6.0 before you do anything else.

How to Find Every Joomla Site Running PageBuilder CK with mySites.guru

When a flaw drops in a component that could be on dozens of your client sites, the first question is always the same: which of my sites run it, and on what version? If you manage 50 or 200 Joomla sites, logging into each admin to check the PageBuilder CK version is not realistic. The bots do not wait for you to finish the list.

mySites.guru records the exact version of every installed extension across every connected Joomla site on a twice-daily snapshot. The extension search shows you every site running PageBuilder CK, grouped by version, in seconds. Filter for anything below 3.6.0 and you have your work list.

View all your PageBuilder CK installations

Open PageBuilder CK Extension Search

Lists every installed version across all your connected Joomla sites. Filter for anything below 3.6.0 to find the installations that need updating.

Combined with the mass extension updater, you can push 3.6.0 across every affected site in one batch instead of a day of logging into admin panels one at a time. What would have been a frantic afternoon becomes a triage you finish over a coffee.

If you do not have a mySites.guru account yet, start a free month and connect your sites. The extension index builds automatically on the first snapshot, and you will know straight away which sites are exposed.

What the Bug Actually Did

PageBuilder CK does its work through a set of front-end endpoints, the kind of background requests a builder makes while you drag elements around and save your layout. One of those endpoints handles file uploads, for adding images to a page.

The problem is what that endpoint did not check. Before a site does anything sensitive, it should ask two separate questions: who is this (authentication), and are they allowed to do this (authorization). The upload endpoint asked neither. The only thing it checked was a Joomla anti-CSRF token, a value meant to confirm a request came from a real page on the site rather than being forged from somewhere else. A CSRF token is not a login. It does not prove who you are, and it does not prove you have permission. And because Joomla prints that token into its own public pages, any visitor can grab one in a single request.

So a guest, with no account and no session, could reach the upload code. That alone would be bad. What makes this critical is the second half: the endpoint took the destination folder straight from the request and the filename, including its extension, exactly as supplied. There was no allow-list restricting uploads to images, no block on PHP files, and no confinement to a media directory. The caller picked the folder, and the caller picked the file.

Put those together and the attack writes itself. Upload `something.php` into a web-served folder, then load that file in a browser, and the server runs it. That is remote code execution, the most severe class of web vulnerability there is, reachable by anyone on the internet.

> A note on what we are not publishing. We confirmed this flaw by reading the code and reproducing it on our own test install, and we reported nothing that is not already implied by the public fix. We are deliberately not naming the exact endpoint or sharing a working request. The fix is out, and the responsible thing is to get people patched, not to hand a recipe to the next scanner, although with AI help they could replicate what we did in seconds.

Why “Just Block /images” Is Not Enough

The instinct with an upload flaw is to stop PHP files from running in the upload folder. For a lot of vulnerabilities that is a reasonable stop-gap. Here it is not, because the attacker is not limited to the upload folder. They name the destination in the request, so if you lock down one folder they simply write to another. In our testing the same upload landed and ran in several different directories, and a planted file could just as easily sit among legitimate admin files where you would never think to look for it.

This is why the real answer is the update, or, if you genuinely cannot update for a few hours, blocking the vulnerable request at a web application firewall or reverse proxy so it never reaches the component at all.

Server-level hardening helps, but know what it does and does not do. If you run Akeeba Admin Tools, its `.htaccess` Maker will not stop the upload itself, because the write happens inside the component, below the level `.htaccess` sees. What its long generated `.htaccess` does do is disable PHP execution in a list of directories, so a dropper that lands in one of those folders cannot be run by requesting it directly. That neutralises a good chunk of the risk. The catch is that the Maker deliberately leaves exceptions where PHP genuinely needs to run, and since this flaw lets the attacker choose any folder, those exceptions are exactly where a working shell could still end up. Helpful, then, but not a substitute for the update. A single `.htaccess` rule scoped to one folder gives an even falser sense of safety against a flaw that can write anywhere.

How to Fix the PageBuilder CK Hack with mySites.guru Easily

Updating is the priority and it is straightforward. In each site's Joomla admin, go to **System** then **Update** then **Extensions**, find PageBuilder CK, and update it to 3.6.0 or newer. If the update does not appear there, download the latest build from joomlack.fr and install it over the top. If you run more than a handful of sites, push the update across all of them at once from your mySites.guru dashboard rather than working through admin panels one by one.

Then check for a break-in, because updating stops the next attempt but does nothing about one that already happened. Look for PHP files that should not exist, and do not only look in the obvious places. Because this flaw lets the attacker choose the folder, a web shell could be sitting in `/images`, `/media`, `/templates`, `/administrator`, or anywhere else writable. While you are in the Joomla admin, check your **Users** list for any Super User accounts you do not recognise and remove them. If you find anything, clean the site properly, change your Joomla passwords and secrets, and run a full security audit across the whole site rather than just the file you found. Whoever gets in through one hole usually leaves a second one somewhere quieter.

mySites.guru helps with the detection half too. Its suspect content tool runs on every twice-daily snapshot across every connected Joomla site and flags known web shell

signatures and files that do not belong. If a site shows a threat in its Hacked? section, treat it as compromised and work through the cleanup.

This Keeps Happening to Joomla Components

PageBuilder CK is not an outlier. It joins a steady run of third-party Joomla component flaws we have written up this year, and several share the exact same shape: an endpoint that should have been locked down, reachable without a login, doing something dangerous. The SP Page Builder zero day from a different vendor was another unauthenticated upload in a page builder, used to plant hidden Joomla admins. The iCagenda zero day was another unauthenticated file upload. The Novarain framework RCE was unauthenticated code execution in a widely-installed extension. The vulnerable JCE editor was, again, a file upload reached without proper checks.

The thread running through all of them is the one we pulled on at length in AJAX endpoints are a big CMS security blind spot: a front-end endpoint that verifies a CSRF token and then stops checking, never asking whether the person behind the request is logged in or allowed to do what they are asking. PageBuilder CK is a textbook case of it, and the 3.6.0 fix is, in essence, adding the permission check that should have been there all along. When a framework leaves authorization up to each individual extension developer, the same mistake gets made independently, over and over.

Joomla's core team has started closing that gap from their side. The recent updates Joomla 5.4.4 and 6.0.4 hardened the core `com_ajax` component, which for years let AJAX handlers in the administrator application be called without an authenticated session. From those versions on, Joomla blocks that by default, and a developer who genuinely needs an open handler has to opt in explicitly. It is a deliberate backwards-compatibility break, made because the safe default was worth more than never breaking anything. It will not retroactively fix a component like PageBuilder CK, whose vulnerable code ran on the public site and did its own routing rather than going through core `com_ajax`, so you still need the 3.6.0 update. But it does mean one large, long-standing version of this mistake is now closed at the source. Keeping your sites on a current Joomla version is part of staying ahead of the next one.

The lesson is not “stop using extensions”. Extensions are what make Joomla useful, and PageBuilder CK is genuinely good at its job. The lesson is that the security of your sites rests on code other people wrote and ship on their own schedule, and the day a flaw like this becomes public you need to know, within minutes, which of your sites run it, patch them all at once, and check whether any were already turned into a foothold. That is the whole reason mySites.guru indexes every extension on every site you connect. When the next one drops, and there will be a next one, you want to be the operator who patched and swept before the scanner came back, not the one finding a strange admin account three weeks later.

The March 2026 Wave That Made the Pattern Obvious

If PageBuilder CK felt like a one-off it would be easy to shrug at. It is not. In a single month, March 2026, at least five separate AJAX and API vulnerabilities landed across Joomla and WordPress, all sharing the same root cause: an endpoint that checks a token but never checks who is calling it. We pulled all five apart in AJAX endpoints are a big CMS security blind spot; here they are at a glance.

- Astroid Framework for Joomla (CVE-2026-21628, CVSS 10.0). The AJAX endpoint verified a CSRF token but never checked the requester was an administrator. Attackers grabbed the token off the public login page and uploaded backdoors. No login required, and a maximum 10.0 score.
- Novarain / Tassos Framework for Joomla (CVE-2026-21627, CVSS 9.5). Fully unauthenticated, no token and no capability check at all. Joomla’s `com_ajax` routed requests to the plugin, which whitelisted file inclusion as a non-admin task, allowing arbitrary PHP inclusion, file deletion and SQL injection.
- Smart Slider 3 for WordPress (CVE-2026-3098, CVSS 6.5). The export AJAX actions had a nonce but no capability check, so any subscriber-level user could read arbitrary files off the server, including `wp-config.php`. Same pattern, different CMS.
- **Joomla core com_ajax ACL hardening** (CVE-2026-21629). The framework that routes every plugin’s AJAX requests was itself missing the default authentication

check in the admin area, the gap that the 5.4.4 and 6.0.4 updates closed.

- **Joomla webservice endpoint access bypass** (CVE-2026-23899). The webservice API did not properly verify permissions on incoming requests, allowing access to endpoints that should have been restricted. A different mechanism, the same category of failure.

Five in one month, three months before PageBuilder CK joined the list. This is not a run of bad luck. It is a structural weakness in how CMS extensions handle authorization, and it is why "which of my sites run the thing that just got patched" needs to be a question you can answer in seconds, not days.

And these are not just disclosures sitting in a database. The same pattern hit JCE, the single most-installed Joomla editor (CVE-2026-48907), where an unauthenticated editor-profile upload let attackers write arbitrary files. That one was actively exploited: we saw it used in the wild to plant rogue JCE editor profiles and drop webshells into `tmp`, `media`, `images` and the `libraries` tree, the exact "endpoint reached without a login, then a file written somewhere it can run" shape as PageBuilder CK. We built a dedicated check that finds the JCE rogue profiles and backdoors and cleans them from one screen, because enough sites were hit that "is JCE installed and patched" was no longer the only question worth asking. When a pattern gets weaponised like that, the gap between disclosure and patching is measured in how exposed your sites are, not in how interesting the bug is.

How mySites.guru Makes Urgent 0-Day Updates Like This Painless

A 0-day is a race, and the clock starts the moment the fix goes public. The slow part has never been the update itself, it is everything around it: working out which of your sites even have the affected extension, on what version, and then getting the patch onto all of them before a scanner gets there first. Done by hand across a portfolio of client sites, that is an afternoon of logging into admin panels. By the time you finish the list, the first sites have been exposed for hours.

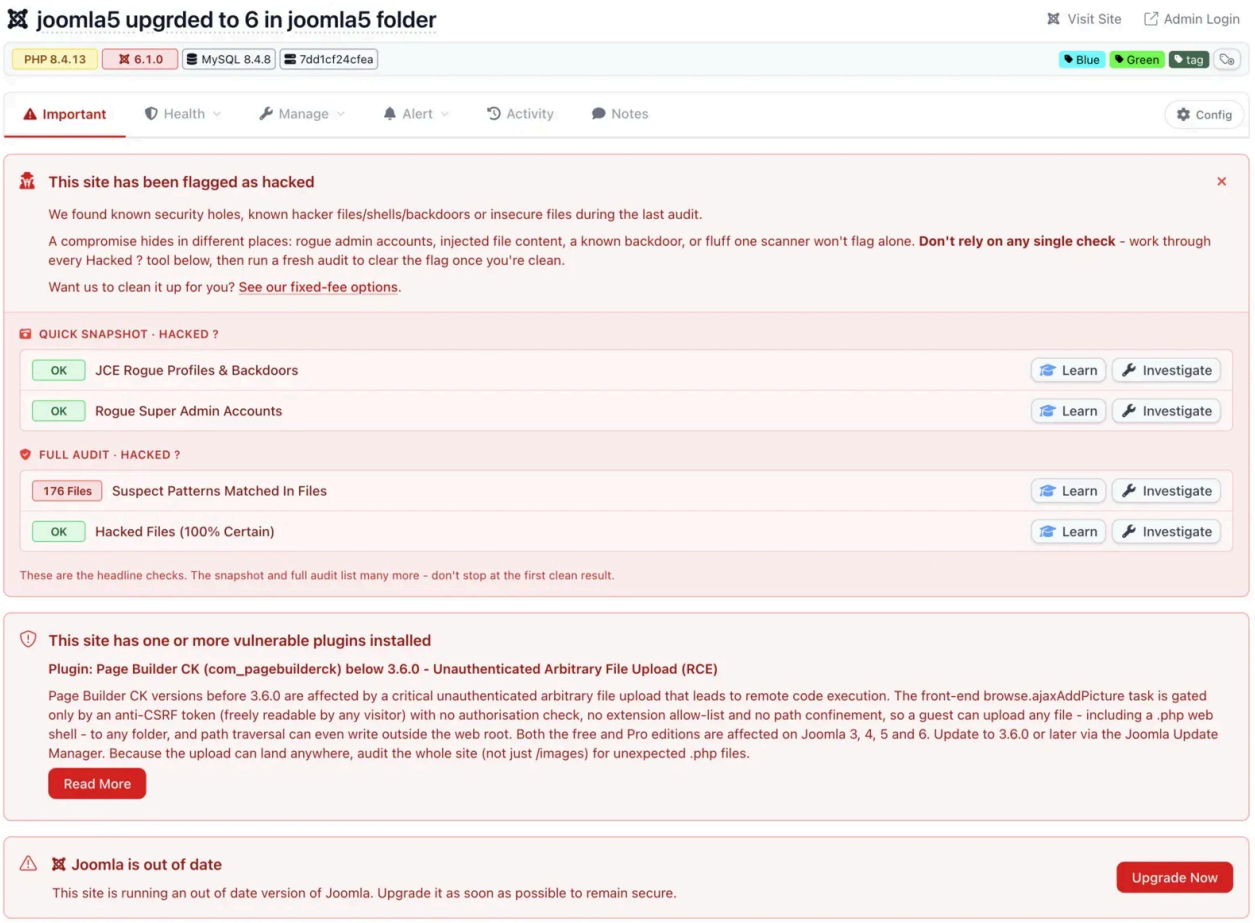
mySites.guru collapses that race into a few clicks, in three steps.

First, **know which sites are affected, in seconds.** Every connected site reports the exact version of every installed extension on a twice-daily snapshot. The extension search indexes that data across your whole portfolio and groups it by version, so a question like “which of my sites run PageBuilder CK below 3.6.0” is answered the moment you ask it, not after a day of checking. This is the same index that let us find 8,297 sites running the Novarain Framework within hours of that disclosure.

Second, **patch all of them at once.** Once you have the list, the mass update tool pushes 3.6.0 to every affected site in a single operation. Whether that is five sites or five hundred, it is one action instead of hundreds. If a site is somehow missing the update in its own admin, you can mass install the extension straight from the dashboard. The full day of manual patching becomes a job you finish over a coffee.

Third, **confirm nobody beat you to it.** Updating closes the door, but on a fast-moving 0-day you also want to know whether anything got in before you patched. The suspect content tool runs on every snapshot, twice a day on every connected Joomla site, flagging web shells and files that do not belong, including in the odd corners this particular flaw can write to.

We have already done two things for this specific flaw. Every connected Joomla site running PageBuilder CK below 3.6.0 is now flagged on the new **Important** tab, the red triage view that surfaces a site’s most urgent problems in one place. It names the component, explains the risk, and links straight to this write-up:



And until active exploitation appears and we ship a dedicated check for it, the suspect content tool already has your back. It reads every file in your webspace and flags anything that looks like a backdoor or web shell on its own merits, without us having to handcraft a pattern for this particular attack first. That is the difference between a tool that only knows the threats someone has already written a rule for and one that can catch a shell it has never seen. It does flag broadly, so it is worth knowing how to read a suspect content match versus a confirmed hacked file before you act on what it raises. When we do start recording sites hit through this flaw, we will add a targeted check that hunts its specific fingerprint, exactly as we did with the JCE rogue profiles and backdoors tool.

That is the whole point of connecting your sites before the next 0-day rather than after. When it drops, you are not scrambling to build a list. You already have it, you push one update, and you move on. If you are not set up yet, start a free month and the index builds itself on the first snapshot.

CVE Record

No CVE has been assigned to this vulnerability as of publication. It was disclosed by the vendor through the 3.6.0 release rather than through a formal advisory, and the changelog gives no technical detail. We will update this post with a CVE identifier if one is published.

Field	Detail
Component	PageBuilder CK (`com_pagebuilderck`)
Vendor	Cedric Keiflin (joomlack.fr)
Type	Unauthenticated arbitrary file upload to remote code execution
Affected versions	Up to and including 3.5.10 (current line); older Joomla 3 and Joomla 4 builds below the back-ported patches below
Fixed in	3.6.0 for current Joomla, plus back-ported 3.1.1 (Joomla 3) and 3.4.10 (Joomla 4). All released 27 June 2026
CVE	None assigned at time of writing

This is a developing story. We published within hours of the fix, so expect updates as a CVE is assigned and as we learn more. If and when we see this flaw exploited in the wild, we will ship a dedicated mySites.guru check to find affected sites and report it here. Until then: update to 3.6.0 on every site that runs PageBuilder CK, watch the Important tab, and let the suspect content tool catch anything that slipped in before you patched.

Further Reading

- PageBuilder CK by Cedric Keiflin
- Joomlack forum: new PageBuilder CK release and the Joomla 3 / Joomla 4 back-ports (vendor announcement)

- Joomlack on Facebook (where the vendor posted the patched-version download links)
- Joomla Vulnerable Extensions List
- OWASP: Unrestricted File Upload
- Joomla security best practices

Frequently Asked Questions

What is the PageBuilder CK vulnerability?

PageBuilder CK is a free drag-and-drop page builder for Joomla by Cedric Keiflin. In versions up to and including 3.5.10, one of the component's front-end endpoints accepts a file upload with no authentication and no permission check, and it lets the caller choose which folder the file lands in. An attacker with no login can upload a PHP file into a web-served folder and then run it, which is remote code execution: full control of the site. The fix is PageBuilder CK 3.6.0, released 27 June 2026. Anything below 3.6.0 should be treated as vulnerable.

Which PageBuilder CK versions are affected?

Every version up to and including 3.5.10. We confirmed the flaw by comparing the 3.5.10 and 3.6.0 source and by reproducing the upload end to end on a clean Joomla 5 test install. The main fix is 3.6.0 for current Joomla, released 27 June 2026. The vendor also back-patched the older branches for sites that cannot move forward: PageBuilder CK 3.1.1 for Joomla 3 and 3.4.10 for Joomla 4. So 'below 3.6.0' is the right line for a Joomla 5 or 6 site, but a Joomla 3 site needs at least 3.1.1 and a Joomla 4 site at least 3.4.10. The 3.6.0 changelog simply reads 'IMPORTANT : Fix security issue' with no further detail, which is normal for a quiet security release. No CVE has been assigned yet.

Does this need a login or any special setup to exploit?

No. The upload endpoint runs on the public front end of the site and is gated only by a Joomla anti-CSRF token, which any visitor can read straight off the site's own pages. There is no check for who the user is or whether they are allowed to upload files. The component just needs to be installed and enabled. There is no menu item, no published page, and no permission toggle required for the flaw to work.

How do I fix it?

Update PageBuilder CK to 3.6.0 or later on every Joomla site that runs it. You can update through the Joomla admin under System then Update then Extensions, push the update across many sites at once from your mySites.guru dashboard, or download the latest build from joomlack.fr and install it over the top. Updating closes the door, but it does not undo a break-in that already happened, so also check each affected site for files that should not be there.

Does unpublishing PageBuilder CK protect the site?

No. Unpublishing the component, or unpublishing a page that uses it, does not close the hole. The vulnerable endpoint is still reachable and the files it writes are still web-served. Only updating to 3.6.0, or blocking the endpoint at a firewall, actually stops it. Because the flaw lets the attacker write to any folder, blocking PHP execution in one folder like /images does not contain it either.

How do I know if one of my sites was already hacked?

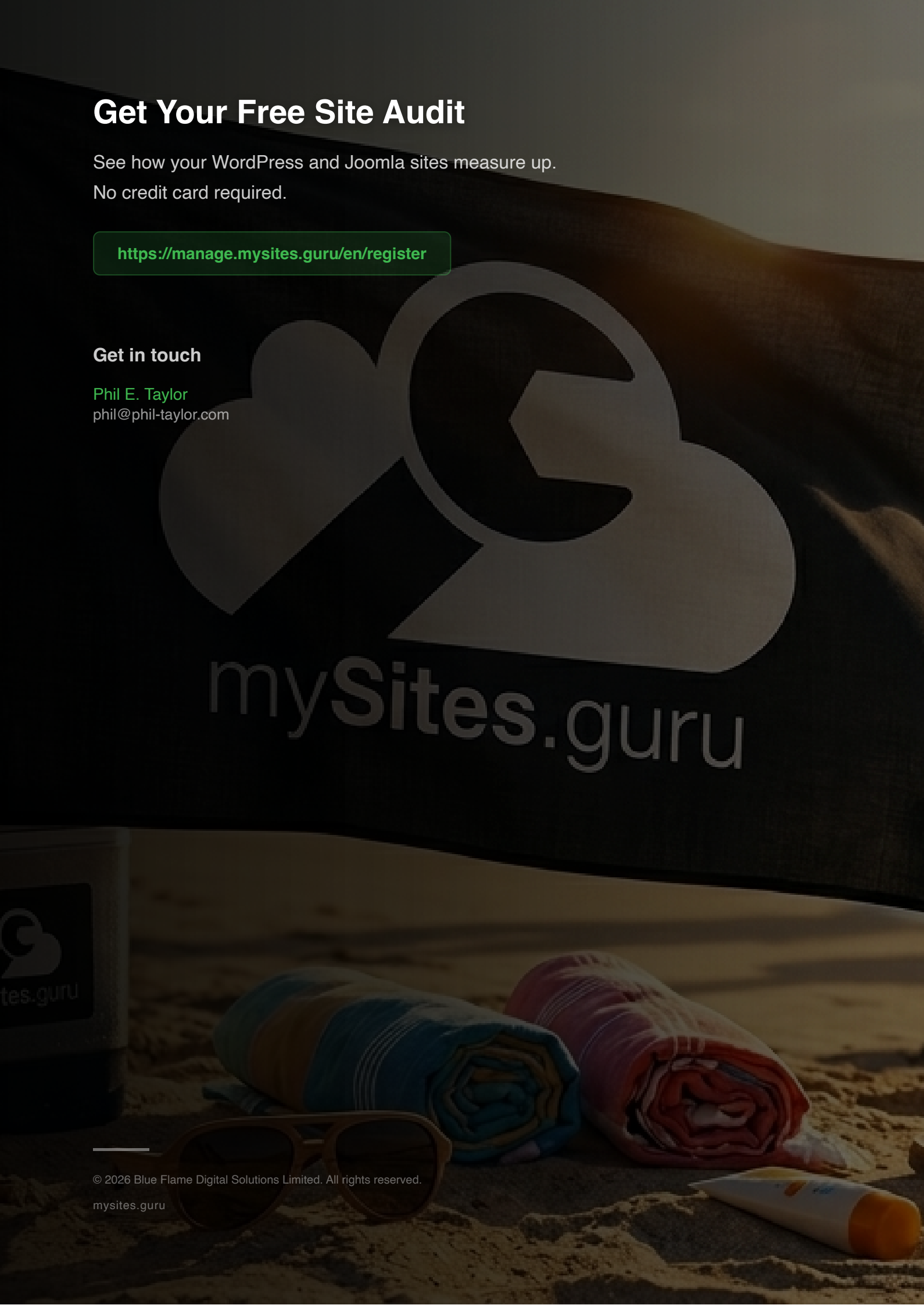
Because this flaw lets the attacker pick the destination folder, a planted file could be anywhere, not just the obvious upload directories. Look for PHP files that have no business existing under /images, /media, /templates and /administrator. the mySites.guru suspect content tool runs on every snapshot, twice a day, on every connected Joomla site, and flags known web shell signatures and files that do not belong. If a site shows a threat in its Hacked? section, treat it as compromised, clean it properly, then change your Joomla passwords and secrets.

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