mySites.guru

Zero Day Vulnerability Found in iCagenda Joomla Extension

We found and confirmed an unauthenticated file upload that gave attackers full remote code execution on any Joomla site running iCagenda. It was already being exploited. The developer shipped 4.0.8 the same day.

Phil E. Taylor | 15 June 2026

JCE Profiles Hack (15th June): Attackers are actively targeting Joomla sites that run JCE. Track down the rogue profiles and webshells they leave behind, on every site you manage.

Read the alert >

A client forwarded us an access log on a Sunday morning with one line that mattered. A bot, calling itself `icagenda-batch/1.0`, had posted something to their Joomla site and then immediately gone looking for a `.php` file in the events component's upload folder. That is the shape of an attack that already knows it has won. By the end of the day we had confirmed a critical zero day in iCagenda, reproduced it from start to finish, reported it to the developer, and the developer had shipped a fix.

iCagenda is a popular events and calendar component for Joomla, by JoomliC. The flaw let anyone, with no login at all, upload a PHP web shell through the public event submission form and then run it. That is remote code execution: full control to steal data, deface pages, plant backdoors, or use the server to attack other people. It affected every version up to and including 4.0.7, which was the latest version that morning, and it was already being exploited in the wild. That last part is what makes it a zero day rather than a quiet finding.

The patch is iCagenda 4.0.8, released on 15 June 2026. If you run any Joomla site with iCagenda installed, update it now, then read on for what the bug actually did and how to check whether you were already hit.

TL;DR

- **Unauthenticated file upload to remote code execution** in iCagenda's frontend event submission form. No login required
- Affects **every version up to and including 4.0.7**. Fixed in **4.0.8**, released 15 June 2026
- **Already exploited in the wild** when we found it, by an automated scanner identifying as `icagenda-batch/1.0`

- We confirmed it by code review, reproduced it end to end, and sent the developer a safe proof of concept. **JoomliC shipped 4.0.8 the same day**, and we reviewed the new code to confirm the fix is real
- **Update to 4.0.8 on every affected site, then check for compromise.** Unpublishing the component does not protect you

This is a fast-moving disclosure. The fix exists, but the full technical detail is going public now, which means opportunistic scanners will know exactly what to look for. The sites that get caught are the ones still on an old version when that happens. Update to 4.0.8 before you do anything else.

How to Check Your Joomla Sites for iCagenda with mySites.guru

When a zero day drops in a component that could be on dozens of your client sites, the first question is always the same: which of my sites are running it, and on what version? If you manage 50 or 200 Joomla sites, logging into each admin to check the iCagenda 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 iCagenda, grouped by version, in seconds. Filter for anything below 4.0.8 and you have your work list.

Combined with the mass extension updater, you can push 4.0.8 across every affected site in one batch instead of a day of logging into admin panels one at a time. A zero day becomes a triage you finish over a coffee.

If you do not have a mySites.guru account yet, start a free trial 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

The vulnerability lived in iCagenda's frontend "Submit an Event" form, the feature that lets visitors propose events for the calendar. The form has a file attachment field. The processing code took the uploaded file, kept the file extension the visitor supplied, and wrote it straight to a folder under the web root:

```
images/icagenda/frontend/attachments/your-file.php
```

There was no allow-list of permitted extensions, no block on `.php`, no check that the file was actually the image type it claimed to be. Joomla ships helpers that do exactly these checks, and the component simply did not call them on this path. So you could upload `shell.php`, browse to it, and the server would run it. That is the whole chain, and it is as bad as web vulnerabilities get.

There was a second half to the bug that made it worse. iCagenda has a setting that controls who is allowed to submit an event, and by default it is set to "Registered", meaning logged-in users only. You would reasonably assume that setting protected the upload. It did not. The access check was applied only in the view that decides whether to draw the form on screen. The controller that actually processed the submission and wrote the file never checked it at all.

That means an attacker did not need an account, and did not even need the submit form to be publicly linked anywhere. They harvested a form token from any public iCagenda page, such as the events list, and posted their upload straight to the processing endpoint. The "Registered only" setting was bypassed completely. A site with no public submit menu at all was still exploitable.

> We are deliberately not publishing a copy-and-paste exploit. The mechanism above is enough to understand the risk and confirm your sites are patched. It is not a recipe. The fix is out, so the right response to this post is to update, not to test it on someone else's site.

We Saw It Being Used Before There Was a Fix

This was not found by reading code in the abstract. The first evidence was three lines in a client's real access log, in sequence: grab a session, post the malicious upload to the submit endpoint, then fetch the planted shell at the exact path the component writes attachments to. The user agent was `icagenda-batch/1.0`, which tells you this is tooling built specifically to grind through lists of Joomla sites running iCagenda.

That sequence is the textbook signature of a working exploit, not a probe. It is also why we did not sit quietly and wait for a patch before warning people. A flaw being actively used in the wild, with no fixed version to update to, is the definition of a zero day, and the only responsible move was to get the warning out while pushing the developer for a fix.

To confirm it was the component and not something else on the client's server, we reproduced the whole attack on a clean local Joomla 5 install running the exact 4.0.7 build, using a safe proof of concept that only wrote a harmless marker file. It worked on defaults, with nothing special configured. No guest submission toggle, no special category, nothing. The upload succeeded from an anonymous session because the access check was in the wrong place.

Update to 4.0.8, and Mind the Old Files

The fix is iCagenda 4.0.8. We did not just trust the version number, we read the new code. It properly closes both halves of the bug: the upload now goes through Joomla's own `MediaHelper` allow-list, which rejects `.php` and other dangerous extensions, and the access check is now enforced on the processing path, not only in the view. Both fixes are real. Updating to 4.0.8 is the right move.

You have three ways to do it:

- **Joomla admin:** update iCagenda the normal way through the Joomla updater
- **mySites.guru dashboard:** push the update across several sites at once without logging into each one

- **Direct download:** grab 4.0.8 from icagenda.com and install it over the top

The one thing that matters is that you land on 4.0.8 or later. If you took emergency action before the patch existed and renamed or removed the component, reinstall it from a clean 4.0.8 download rather than putting the old files back, otherwise you are restoring the vulnerable version.

One trap worth calling out: unpublishing iCagenda in the Joomla admin does not protect the site. The submit endpoint stays reachable and the uploaded files stay both writable and web-served regardless of whether the component is published. Before 4.0.8 existed, the only reliable stop short of patching was to rename the `com_icagenda` folders so Joomla could not route to them. Now that there is a real fix, just update.

Updating Closes the Door. It Does Not Undo a Break-In

This is the step people skip, and it is the one that matters most if a site was already targeted. Updating to 4.0.8 stops the next attempt. It does nothing about a shell that was uploaded last week. Because this was being actively exploited before the patch, you have to assume some sites were hit and check.

The thing to look for is any file that should not exist under `images/icagenda/frontend/attachments/`. That folder is only ever meant to hold uploaded event attachments. A `.php` file in there is a web shell until proven otherwise. mySites.guru runs a file scanner on every snapshot, twice a day, on every connected Joomla site, and flags known web shell signatures and files that have no business being there. Open each affected site in the dashboard and check its **Hacked?** section.

If you find something, treat that site as compromised and clean it properly: keep a copy of anything suspicious for evidence, remove it, change your Joomla passwords and secrets, and audit the whole site rather than just the iCagenda folder. Anyone who got in through one component will usually leave a second way back in somewhere you would not think to look. Finding one mess is a reason to check the whole house.

This Keeps Happening to Joomla Components

iCagenda is not an outlier. It joins a steady run of third-party Joomla component vulnerabilities we have written up this year, several with the same shape: an endpoint that should have been locked down, reachable without authentication, doing something dangerous. The Astroid framework backdoor and the Novarain framework RCE were both unauthenticated code execution in widely-installed extensions. The vulnerable JCE editor was another file upload. The underlying pattern, where an AJAX or form endpoint skips its authorization check, is one of the most common ways Joomla sites get hacked through no fault of the site owner.

The lesson is not "stop using extensions". Extensions are what make Joomla useful. The lesson is that the security of your sites depends on code other people wrote, shipped on someone else's schedule, and the day a flaw like this becomes public you need to know, within minutes, which of your sites are exposed and be able to patch them all at once. That is the entire 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 before the scanner arrived, not the one reading their access log afterwards.

Disclosure Timeline

- **15 June 2026:** A client forwards an access log showing live exploitation by `icagenda-batch/1.0`
- **15 June 2026:** We confirm the flaw by code review of 4.0.7 and reproduce it end to end on a clean local Joomla 5 install with a safe proof of concept
- **15 June 2026:** We send a responsible-disclosure report and the safe proof of concept to the developer
- **15 June 2026:** JoomliC releases iCagenda 4.0.8, flagged as a critical security release. We review the new code and confirm both halves of the bug are properly fixed

Credit to JoomliC for turning a same-day fix around on a critical report. That is exactly how responsible disclosure is supposed to go.

Further Reading

- iCagenda official site and download
- Joomla Vulnerable Extensions List
- OWASP: Unrestricted File Upload
- Joomla security best practices

Frequently Asked Questions

What is the iCagenda vulnerability?

An unauthenticated arbitrary file upload that leads to remote code execution. iCagenda's frontend 'Submit an Event' form accepted file attachments with no server-side check on the file type, so a visitor with no login could upload a PHP web shell to a predictable, web-served folder and then run it. That gives full control of the site. It affected every version up to and including 4.0.7 and is fixed in 4.0.8.

Which iCagenda versions are affected?

Every version of the current line up to and including 4.0.7, the version that was the latest before this. We confirmed the flaw by code review and reproduced it end to end on 4.0.7. The fix ships in 4.0.8, released on 15 June 2026. Anything below 4.0.8 should be treated as vulnerable.

Was this being exploited in the wild?

Yes. We first saw it in a client's access log: an automated scanner identifying itself as 'icagenda-batch/1.0' grabbed a token, posted a malicious upload to the submit endpoint, then fetched the planted shell at the exact path the component writes attachments to. This is why we treated it as a zero day and not a theoretical finding. The bug was being used before any patch existed.

How do I fix it?

Update iCagenda to 4.0.8 or later on every site that runs it. You can update through the Joomla admin, push the update from your mySites.guru dashboard across many sites at once, or download 4.0.8 from icagenda.com 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 signs of compromise.

Does unpublishing iCagenda protect the site?

No. Unpublishing the component in the Joomla admin, or unpublishing the menu item, does not close the hole. The submit endpoint is still reachable and the uploaded files stay both writable and web-served. Before 4.0.8 existed, the only reliable stop short of patching was to rename the com_icagenda component folders so Joomla could not route to them at all. Now that 4.0.8 is out, just update.

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

Look for files that should not exist under images/icagenda/frontend/attachments/. That folder should only ever hold uploaded attachments, never a .php file. mySites.guru runs a file scanner on every snapshot, twice a day, on every connected Joomla site, and flags known web shell signatures. If a site shows a threat in its Hacked? section, treat it as compromised and clean it properly, then change your Joomla passwords and secrets.

Did iCagenda need guest submission turned on to be exploitable?

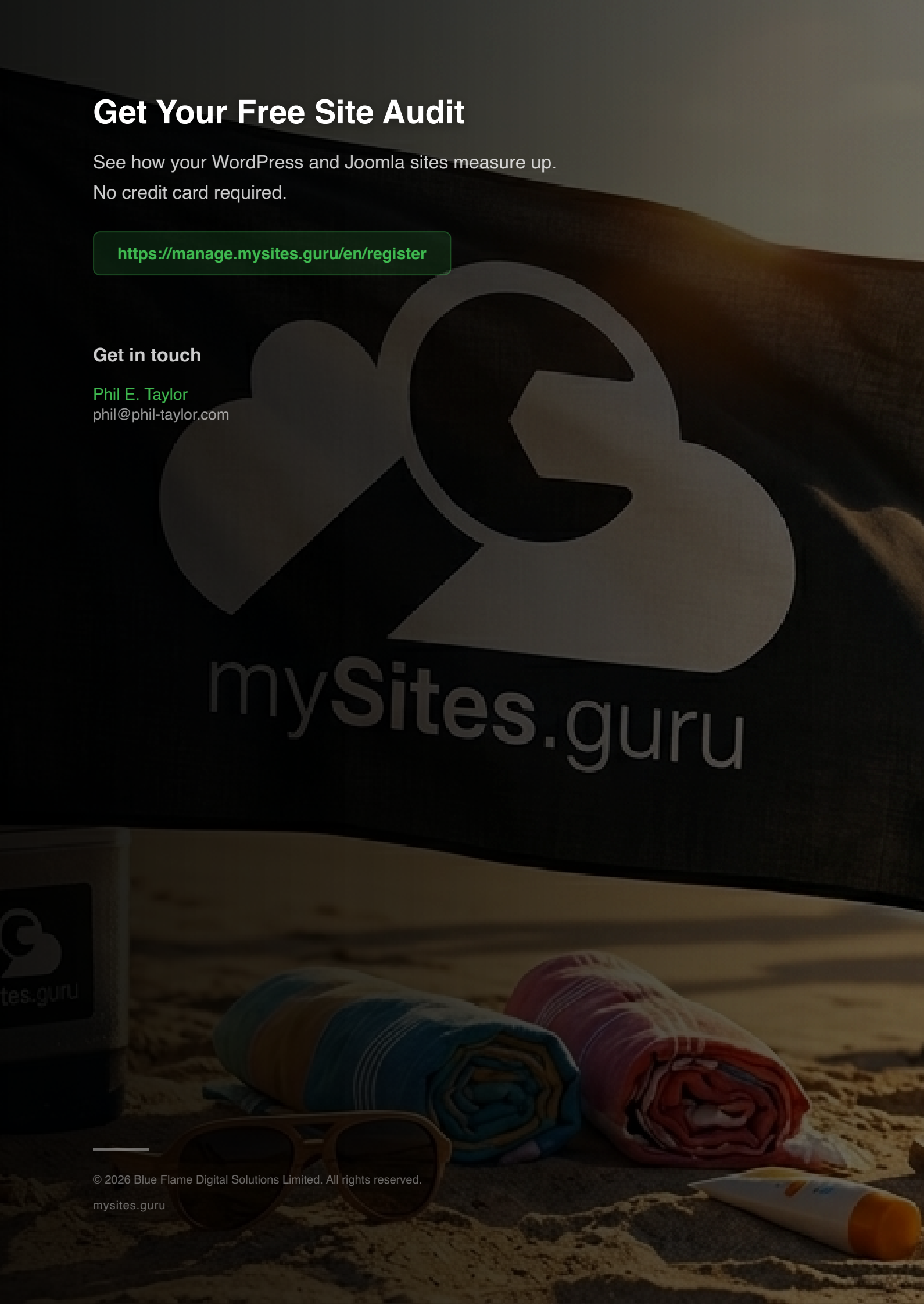
No. The access setting that is supposed to restrict who can submit an event was only checked in the view that renders the form, never on the processing path that handles the upload. So posting straight to the submit endpoint with a token harvested from any public iCagenda page bypassed the setting entirely. A site with the form set to 'Registered only', and even a site with no public submit menu at all, was still exploitable. 4.0.8 enforces the access check on the processing path as well.

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