mySites.guru

Contact Form 7 6.2.4 Fixes an Unauthenticated File Upload in 6.2 to 6.2.3

Contact Form 7 6.2.4 closes an unauthenticated file upload in 6.2 to 6.2.3. Who is exposed, the safelist change that renames docx and zip uploads, what to do.

Phil E. Taylor | 29 October 2026

Rock Lobster released Contact Form 7 6.2.4 on 29 October 2026 to close an unauthenticated arbitrary file upload. It affects Contact Form 7 6.2, 6.2.1, 6.2.2 and 6.2.3, and only those, on forms that have a file upload field. If any of your WordPress sites has the plugin on that line, update it today. The version to install is 6.2.4.1, which followed about two hours later the same day and is the current release on wordpress.org; anything from 6.2.4 upwards has the fix. One thing to know before you click: 6.2.4 changes how uploaded file names are sanitised, and a form that accepts Word documents, spreadsheets or ZIP files will stop keeping them as-is until you add those extensions back with a one-line filter.

Contact Form 7 is Takayuki Miyoshi's plugin, published by Rock Lobster Inc., and wordpress.org counts it on 10,000,000+ WordPress sites.

What the Contact Form 7 6.2.4 security fix covers

Rock Lobster's announcement describes the issue directly: "The issue is an unauthenticated arbitrary file upload. Attackers can leverage multiple flaws in Contact Form 7 6.2 to 6.2.3, in combination, to upload files that should normally be blocked by validation logic and keep files that should normally be removed immediately after submission."

Two conditions narrow who is exposed. The form needs a file uploading field, and, in the vendor's words, "Theoretically, all forms that have file uploading fields and are hosted on an HTTP server configured to list directory contents publicly can be a target." A contact form with only name, email and message fields is not in scope, and a server with directory listing switched off is outside what the vendor describes as a target. Neither of those is a reason to skip the update.

The vulnerable line was the current release for 24 days: 6.2 was released on 5 October 2026, 6.2.1 on 8 October, 6.2.2 on 11 October and 6.2.3 on 21 October. Rock Lobster's position on who should update is broad: "While the vulnerability is seen only

in Contact Form 7 6.2 to 6.2.3, since the security enhancements apply to all versions of the plugin, we recommend upgrading to 6.2.4 for all users."

The issue was found by Shinya Murakami and Jakub Herman and reported through WPScan. No CVE has been assigned yet, it is not in CISA's Known Exploited Vulnerabilities list, and the vendor has not said it is being exploited.

Beyond the security change, the release notes list Functions-PHP 1.1.0, FormDataTree-PHP 2.0.9, SWV-PHP 2.0.8, and `wp_scrub_utf8()` applied to text before UTF-8 regex matching. The v6.2.3 to v6.2.4 compare on GitHub has the full diff. 6.2.4.1's changelog entry reads "FormDataTree-PHP: Upgraded to 2.0.10". Current Contact Form 7 requires WordPress 7.1 or higher and PHP 8.3.

How many WordPress sites were on the affected Contact Form 7 versions?

Over half of them, in the mySites.guru database. We measured the Contact Form 7 installs on the WordPress sites we monitor, snapshotted in the last 30 days, at around midday UTC on 29 October, a few hours after the release.

54.6%

were on the affected 6.2 to 6.2.3 line

47.4% on 6.2.3 alone, the version current for the eight days before the fix

36.8%

were still on 6.1.x, outside the affected range

A further 5.8% were on something older, also outside it

2.9%

had already reached 6.2.4 or later

Measured a few hours after the release

Contact Form 7 installs in the mySites.guru extension inventory of WordPress sites snapshotted in the last 30 days, measured 29 October 2026 around midday UTC. Percentages of that set, not of all WordPress sites.

The shape of that data is unusual for a plugin advisory. Normally the exposed sites are the ones that fell behind. Here the flaw lives only in the newest line, so the sites that applied October's four Contact Form 7 releases promptly, the ones doing everything right, are the exposed ones, and the sites that had not touched the plugin since 6.1.x missed it entirely.

None of that argues for updating less often. Most plugin flaws sit in old versions, not new ones, and the fix for this one exists today because the plugin is actively maintained. The lesson for anyone managing many WordPress sites is that an update is not finished when it is installed, because the version you installed can become the version with the hole. What you need is to see the version on every site, quickly, the day a fix comes out, so that 6.2.3 across a client list is a morning's work rather than a question you cannot answer.

Before you update: Contact Form 7 forms that accept Word documents or ZIP files

The security change in 6.2.4 is in `wpcf7_antiscript_file_name()`, the function that sanitises an uploaded file's name. In 6.2.3 and earlier it worked from a blocklist, stripping extensions that could run on the server: php, phtml, pl, py, rb, cgi, asp and aspx. In 6.2.4 it works from a safelist of extensions that are allowed through, and everything else is converted to a safe alternative.

The default safelist is audio, video and image formats plus pdf and txt. In full: 3gp, 3gpp, 3g2, 3gp2, aac, flac, mid, midi, mp3, m4a, m4b, ogg, oga, wav, x-wav, mka, wax, wma, ra, ram, avi, divx, mp4, m4v, mpeg, mpg, mpe, ogv, mov, qt, webm, flv, mkv, asf, asx, wm, wmv, wmx, avif, bmp, gif, heic, heics, heif, heifs, jpg, jpeg, jpe, png, tiff, tif, webp, ico, pdf, txt.

doc, docx, xls, xlsx, odt, rtf, csv and zip are not on the default list

A file with an extension outside the safelist is not rejected. Its name is rewritten so the final extension has `_.txt` appended: `report.docx` arrives as `report.docx_.txt`, and `cv.zip` as `cv.zip_.txt`. A recruitment form, a quote request form or a support form that

takes attachments will keep working and deliver renamed files, with no error, until someone notices.

The fix is the filter Rock Lobster gives in the announcement. It appends to the default list rather than replacing it:

```
add_filter( 'wpcf7_file_extension_safelist', function ( $safelist ) {
    return array_merge( $safelist, ['zip'] ); // Adding .zip
} );
```

Add only the extensions a form needs. If a CV upload field takes PDF and Word documents, add `doc` and `docx` and stop there. The change keeps the list of what a public form accepts short on purpose, and widening it back to "anything" undoes the fix. Put the filter in a small site-specific plugin or the theme's `functions.php`, test one submission with each file type, and check what arrives in the notification email.

There is a second change that most sites can ignore. `wpcf7_unship_uploaded_file()` now requires its second `$options` argument and takes a new third `$uploaded_files` argument by reference. That only matters to custom code that calls the function directly; if you have never heard of it, you do not call it.

Checking an affected WordPress site

Contact Form 7 holds a submitted file briefly during the submission. In the 6.2.4 source it writes to a temporary folder at `wp-content/uploads/wpcf7_uploads`, inside a randomly named 10-digit subfolder per upload. The plugin writes an `.htaccess` in that folder with `Require all denied`, and a clean-up routine removes files older than 60 seconds. That `.htaccess` is an Apache rule. nginx ignores `.htaccess` files, so on nginx the folder is as reachable as its permissions and the server config allow.

The vendor's description of the issue is of files that should have been blocked getting through and files that should have been removed being kept, so that folder is where to look on a site that spent time on 6.2 to 6.2.3 with a file upload field. From the site root:

```
find wp-content/uploads/wpcf7_uploads -type f \
  \( -iname '*.php' -o -iname '*.phtml' -o -iname '*.phar' \) -print
```

A normal site returns nothing, because anything legitimate there was deleted within a minute of arriving. Any script file that comes back should not exist, and the right response is to treat the site as possibly compromised rather than only out of date. Our post on finding hacked files and backdoors covers what to look for across the rest of the webspace, because a dropped file is rarely left where it arrived. If you find something and would rather hand it over, fix.mySites.guru cleans a hacked site for a single fixed fee of £120 per incident, usually same-day.

Then close the condition the vendor named. Directory listing should be off on every site you run, affected or not. On Apache, in the virtual host or the site's root `.htaccess`:

```
Options -Indexes
```

On nginx, in the `server` or `location` block:

```
autoindex off;
```

The nginx default is already off, but an inherited config can switch it on, so check rather than assume. Open `https://example.com/wp-content/uploads/` in a browser: a 403 or a blank page is what you want, and a list of years and months means listing is on.

Find files that can accept uploads

mySites.guru checks every connected site for this automatically and flags it the moment it appears. It runs as part of the full audit on every connected site.

Check your site for free →

Finding every Contact Form 7 install with mySites.guru

The slow part of a release like this is the inventory, and for an agency it is slower than the fix. Logging into each wp-admin, opening Plugins, reading the Contact Form 7 version, and then remembering which forms on which sites have a file field is an afternoon for twenty sites and a week for two hundred, and by the end of it the first sites you checked are stale.

The mySites.guru plugin inventory lists every plugin on every connected WordPress site with the version each one runs. Search for Contact Form 7 once and you have every site and version on one screen, which turns “which of our sites are on 6.2 to 6.2.3?” into a question with an immediate answer. From the same place you can update plugins across many sites from one dashboard; the walkthrough of mass-upgrading Joomla and WordPress sites from one dashboard shows the flow. For the next release, mySites.guru also alerts on known-vulnerable plugin versions, so the question gets asked for you rather than by a customer.

The audit side does the other half of this post’s work. Every connected site gets the full audit, which includes the check above for files capable of accepting an uploaded file, so a stray upload handler on a site you had forgotten about shows up without anyone remembering to look. If you want to see what that looks like on one of your own sites first, a free audit runs the same checks and shows the plugin inventory for that site.

The data above is the case for it: the sites exposed this time were the ones keeping up, and what would have made 29 October a calm day for their owners was being able to see every version in one place within minutes of the announcement. That visibility, plus the bulk update and the alerting for the next one, is what a mySites.guru subscription buys, and the free audit is the way to try it on a site before you pick a plan.

Further Reading

- Contact Form 7 6.2.4 announcement - Rock Lobster's description of the issue, the safelist change and the filter.
- Contact Form 7 v6.2.3 to v6.2.4 on GitHub - the full diff between the last affected version and the fix.
- Contact Form 7 on wordpress.org - current version, changelog and requirements.
- WPScan - the service the issue was reported through.

Run a free audit to see every plugin on a WordPress site, with its version, and the files that can accept an upload.

Frequently Asked Questions

Which versions of Contact Form 7 have the file upload vulnerability?

Contact Form 7 6.2, 6.2.1, 6.2.2 and 6.2.3, released between 5 and 21 October 2026. The vendor says the issue is seen only in that line, but recommends 6.2.4 or later for every install because the security changes apply to all versions. Sites still on 6.1.x or older are outside the affected range.

Is the Contact Form 7 vulnerability being exploited?

Not as far as anyone has said. No CVE has been assigned yet, it is not in CISA's Known Exploited Vulnerabilities list, and Rock Lobster has not reported exploitation. The vendor describes it as an unauthenticated arbitrary file upload, which is serious enough to update for without waiting for that to change.

Should I install Contact Form 7 6.2.4 or 6.2.4.1?

6.2.4.1, which is the current version on wordpress.org. It followed 6.2.4 by about two hours on the same day and its changelog lists one change, FormDataTree-PHP upgraded to 2.0.10. Anything from 6.2.4 upwards contains the security fix.

Why did my form stop accepting Word documents after updating Contact Form 7?

Because 6.2.4 replaced the blocklist of dangerous extensions with a safelist of permitted ones, and the default list covers audio, video and image formats plus pdf and txt. An extension that is not on the list gets _.txt appended, so report.docx becomes report.docx_.txt. Add the extensions your form needs with the wpcf7_file_extension_safelist filter.

Where does Contact Form 7 store uploaded files?

In a temporary folder at wp-content/uploads/wpcf7_uploads, inside a randomly named 10-digit subfolder per upload. The plugin writes an .htaccess with Require all denied there, which only Apache honours, and a clean-up routine removes files older than 60 seconds. Look there for anything with a script extension on a site that was on 6.2 to 6.2.3.

Who found the Contact Form 7 file upload issue?

Shinya Murakami and Jakub Herman, who reported it through WPScan. Rock Lobster credits them in the 6.2.4 release announcement.

How do I find every WordPress site running an affected Contact Form 7 version?

The mySites.guru plugin inventory lists every plugin on every connected WordPress site with the version each one runs, so one search for Contact Form 7 shows every site and version. From there you can update the plugin across many sites from one dashboard. A free audit covers the inventory on one site.

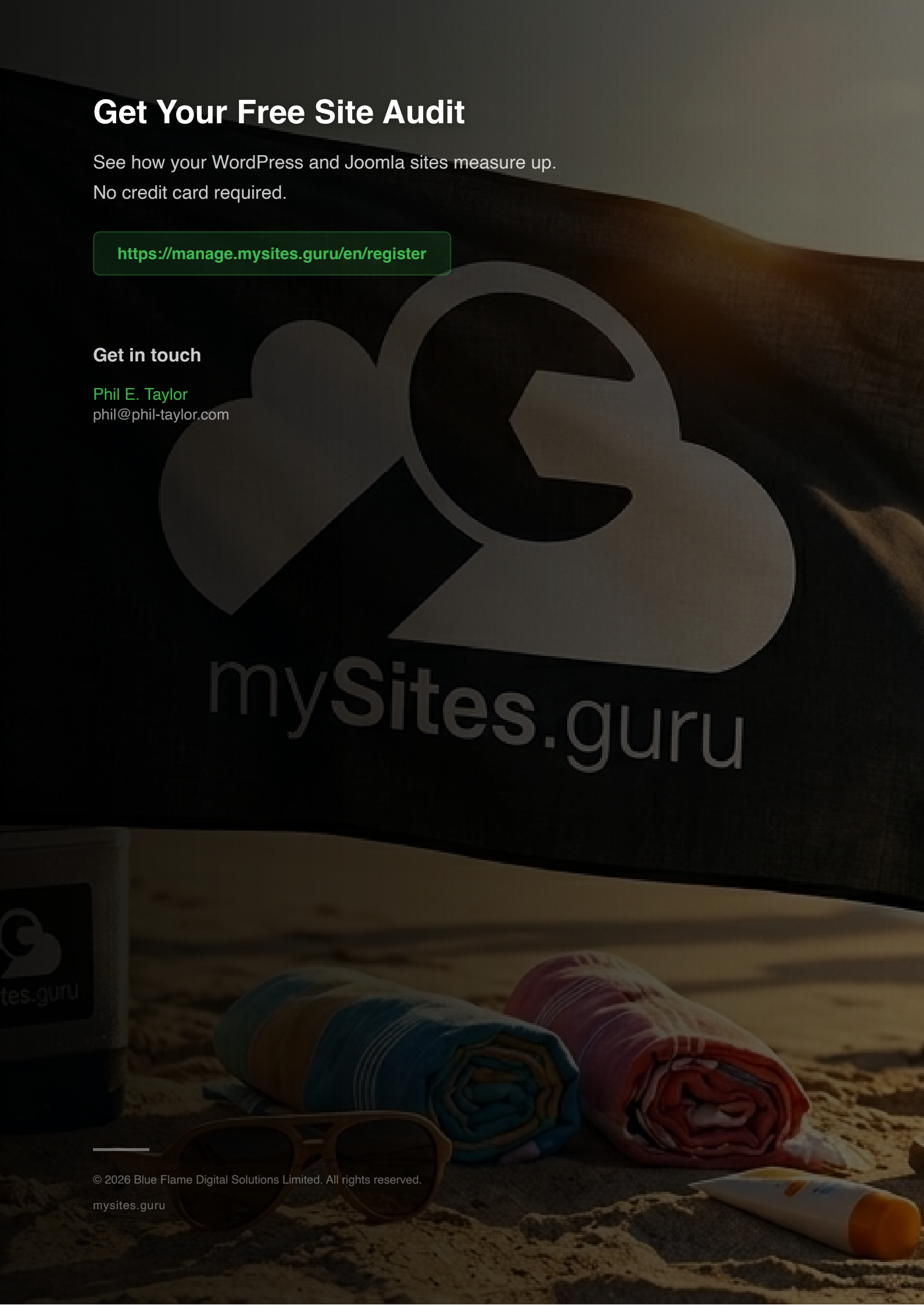
Written and edited by a human, with AI assistance. Our approach to AI

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.