Suspect Content vs Hacked Files: Which Flags Actually Matter in mySites.guru

Your audit flagged dozens of files as suspect content. Most need a look, not a panic. Learn which flags are pattern matches and which are confirmed hacks.

Phil E. Taylor | 11 June 2026

Your audit finishes, you open the Suspect content tool, and there they are: thirty-odd files flagged across your site. Before you start deleting things or drafting a difficult email to your client, look closer at the list. It contains two completely different signals, and knowing which is which changes what you do next.

Most of those rows are pattern matches: files worth a look, nothing more yet. But if any row carries a red **Hacked File** badge, that one is not a maybe. This post explains the difference between the two, and the fastest way to work through the rest of the list. If your sites aren't connected yet, a free security audit will show you exactly this view of your own webspace.

One list, two very different signals

Every mySites.guru audit reads every file in your webspace, every line, including the dormant files a browser-based scanner never sees. The full mechanics of the scan are a post of their own. What matters here is that two separate detection methods feed the one list you're looking at:

- **Pattern matches**: the file contains code matching one or more of 2,000+ hand-written hack patterns. The file is labelled suspect. Translation: worth a look.
- **Hash matches**: the MD5 hash of the entire file matches a previously confirmed hacked file. The file gets the red Hacked File badge. Translation: confirmed.

The first is a heuristic. The second is a fact. Treating them the same wastes hours on files that were never a threat, or worse, leaves you debating a file that has already been condemned.

What does a suspect content match actually mean?

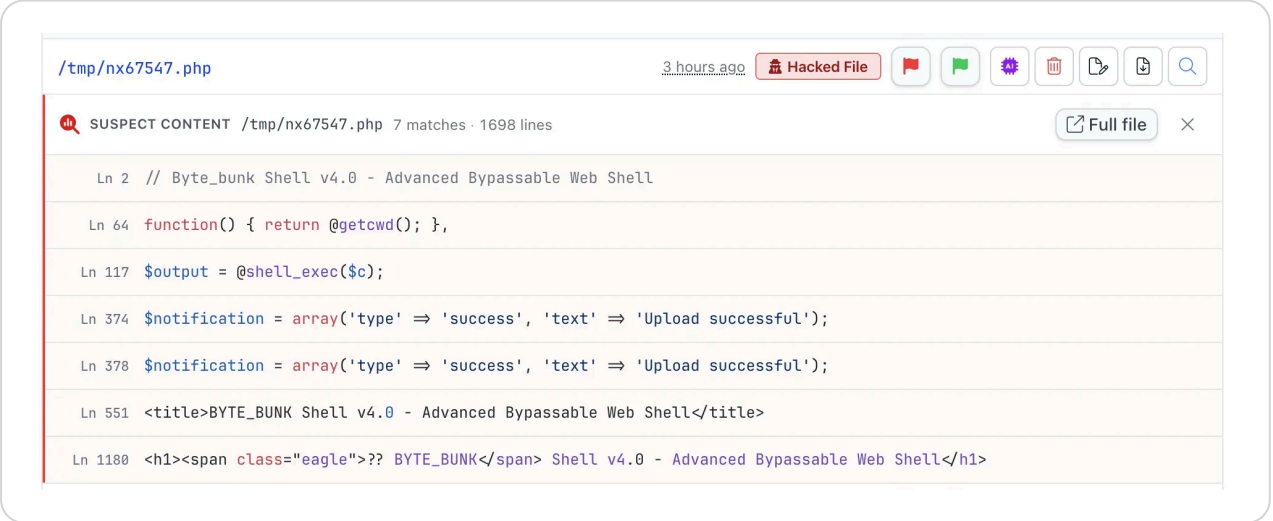
A suspect content match means a file in your webspace contains code matching at least one of more than 2,000 patterns built from real hacks found on real Joomla and

WordPress sites over two decades. It does not mean the file is malicious. It means the file uses code that hackers also use, and it deserves a look.

Hackers write in the same language as legitimate developers. `move_uploaded_file()` powers every legitimate upload form, and every dropped webshell. `mail()` sends your contact form enquiries, and a spammer's payload. Base64 turns up in CSS image handling as often as it does in obfuscated backdoors. A pattern match without context proves nothing either way, which is exactly why the tool calls these files suspect rather than hacked.

False positives here are the design working, not failing. The patterns are deliberately inclusive because the alternative is missing the one real backdoor among the 20,000 files on a typical site. The tool's job is to shrink 20,000 files down to a dozen worth reading, and across the 80,000+ sites connected to mySites.guru, that's exactly what it does.

Click any flagged file name and the tool shows you the exact matched lines with line numbers, alongside the file's modification date, size, and permissions. You can edit the file in place or delete it without ever opening an FTP client. Here's a real one, a BYTE_BUNK webshell caught with 7 pattern matches across its 1,698 lines, from `shell_exec()` calls to its own boastful title tag:

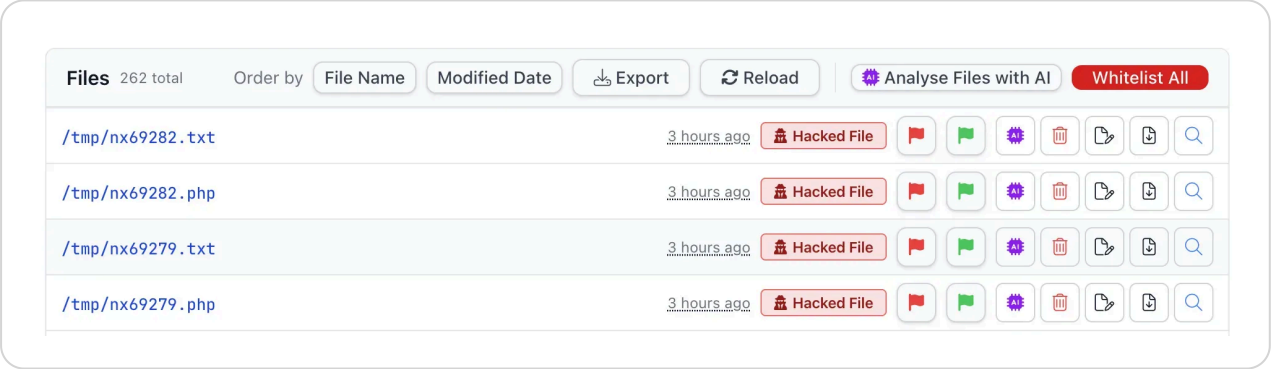


The red Hacked File badge means act now

When a file carries the red Hacked File badge, the MD5 hash of its entire contents matches a file already confirmed as hacked. An identical hash means identical bytes: your file is the same, character for character, as one that has been examined and condemned before. There is no maybe at this confidence level, and no false positives.

Every one of the 14,000+ hashes on that list has been through review before it earned its place. For years that reviewer was me, reading each backdoor line by line before flagging it. Today the AI malware analysis adds confirmed-malicious verdicts to the same list, and I still see every AI verdict and overrule the wrong ones before they pollute the data.

> **Important**
>
> A Hacked File badge means your site is compromised, not just that one file. The attacker got in somehow. Deleting the file removes a symptom, not the entry point.



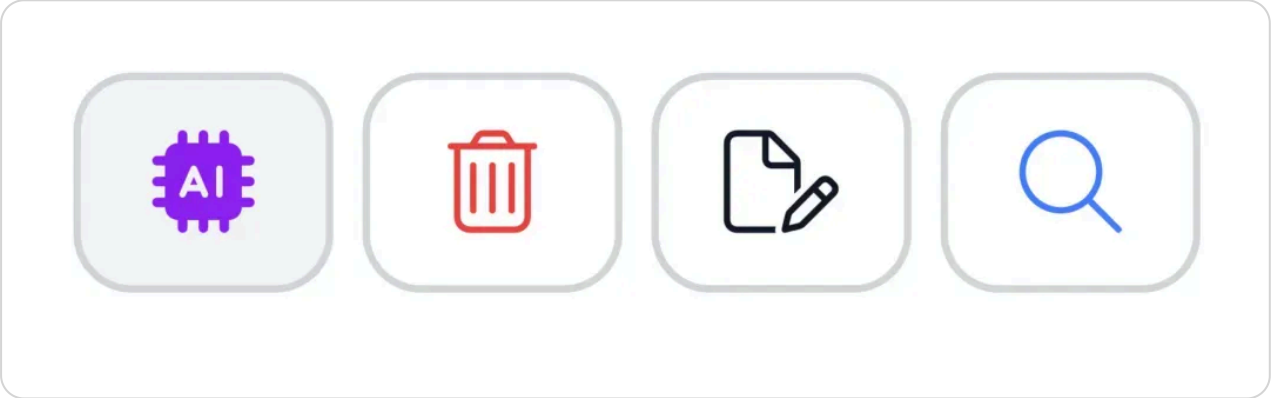
If you see the badge, switch from triage to incident response: follow our guide to fixing a hacked Joomla or WordPress site, or hand the whole job to fix.mysites.guru for a set-fee cleanup.

How do I triage a wall of flagged files?

Work the list in confidence order: confirmed hacks first, AI verdicts second, manual review last. This turns a wall of thirty flags into three short queues, and most of the time the third queue is empty by the time you reach it.

Step 1: Hacked File badges. Anything with the red badge is a confirmed compromise. Stop triaging and start cleaning, as above. The list sorts confirmed files to the top for exactly this reason.

Step 2: AI analysis for everything else. Click the AI icon on any single file, or use Analyse Files with AI to queue the lot. Each file comes back with a verdict: safe, suspicious, or malicious, with the problem lines identified. Verdicts are cached globally by file hash, so the files of a popular plugin have usually been analysed by someone already and come back instantly at no cost.



Step 3: manually review what's left. For the handful of files the AI marks suspicious, open the matched lines and apply context the AI may lack. Does the file belong to a plugin you actually installed? Does its modification date match the rest of the package, or did it change last Tuesday at 3am? Is it a PHP file sitting in an images folder? Compare it against the original extension package if you can.

If a file survives all three steps and you're still unsure, send it to me and I'll look at it myself. That offer has been part of the service since 2012.

Your confirmed hack protects everyone else

Every confirmed hacked file makes the whole network safer. The moment a hash is confirmed, whether by my review or an AI verdict I've checked, it joins the global hacklist, and the next audit of any connected site checks against it. The same backdoor dropped on a hundred sites only needs to be condemned once.

This crowd-sourced model is why detection improves daily rather than quarterly. We run over 3,000 audits a day and find over 200 hacked sites a week, and every one of those finds feeds patterns and hashes back into the checks that run on your sites tomorrow.

What if I'm still not sure about a file?

Don't whitelist it, because you can't: mySites.guru deliberately has no user whitelisting, for reasons (including an expensive legal one) covered in the suspect content deep-dive. A file you hide from yourself today is a backdoor you won't see next month.

Instead, leave it flagged and treat it as a known quantity. A suspect flag costs you nothing while it sits there; you'll re-confirm it in seconds on the next audit with cached AI verdicts. And if a file ever nags at you, that's what the manual review channel is for.

Further Reading

- NSA & ASD: Mitigating Web Shells - joint guidance on detecting and blocking web shell malware, with detection scripts.
- PHP Manual: eval() - the official documentation, complete with its famous warning about why this construct is dangerous.
- WordPress.org: FAQ - My site was hacked - the official WordPress recovery checklist.
- OWASP: Code Injection - background on the attack class most webshells belong to.

Frequently Asked Questions

What is the difference between a suspect file and a hacked file in mySites.guru?

A suspect file matched one or more of our 2,000+ hand-written hack patterns and needs context to judge. A file with the red Hacked File badge has an MD5 hash identical to a previously confirmed hacked file, so it is the same file, byte for byte, and there is no doubt.

Can a file with the red Hacked File badge be a false positive?

No. The badge only appears when the MD5 hash of the entire file matches a hash from a file already confirmed as hacked. Identical hash means identical contents, so the verdict carries over with certainty.

Should I delete every file flagged as suspect content?

No. Many suspect files are legitimate code that happens to use the same PHP functions hackers use. Investigate first: read the matched lines, run the AI analysis, or compare the file against the original plugin or extension package.

Why does mySites.guru flag legitimate files at all?

The patterns are deliberately inclusive because malware authors write in the same language as legitimate developers. Casting a wide net means reviewing a dozen files instead of missing the one that matters among 20,000.

How does the AI malware analysis help with triage?

One click sends a flagged file to Claude or GPT using your own API key and returns a safe, suspicious, or malicious verdict with the problem lines identified. Results are cached globally by file hash, so files already analysed by anyone return instantly at no cost.

What should I do first if I see a Hacked File badge?

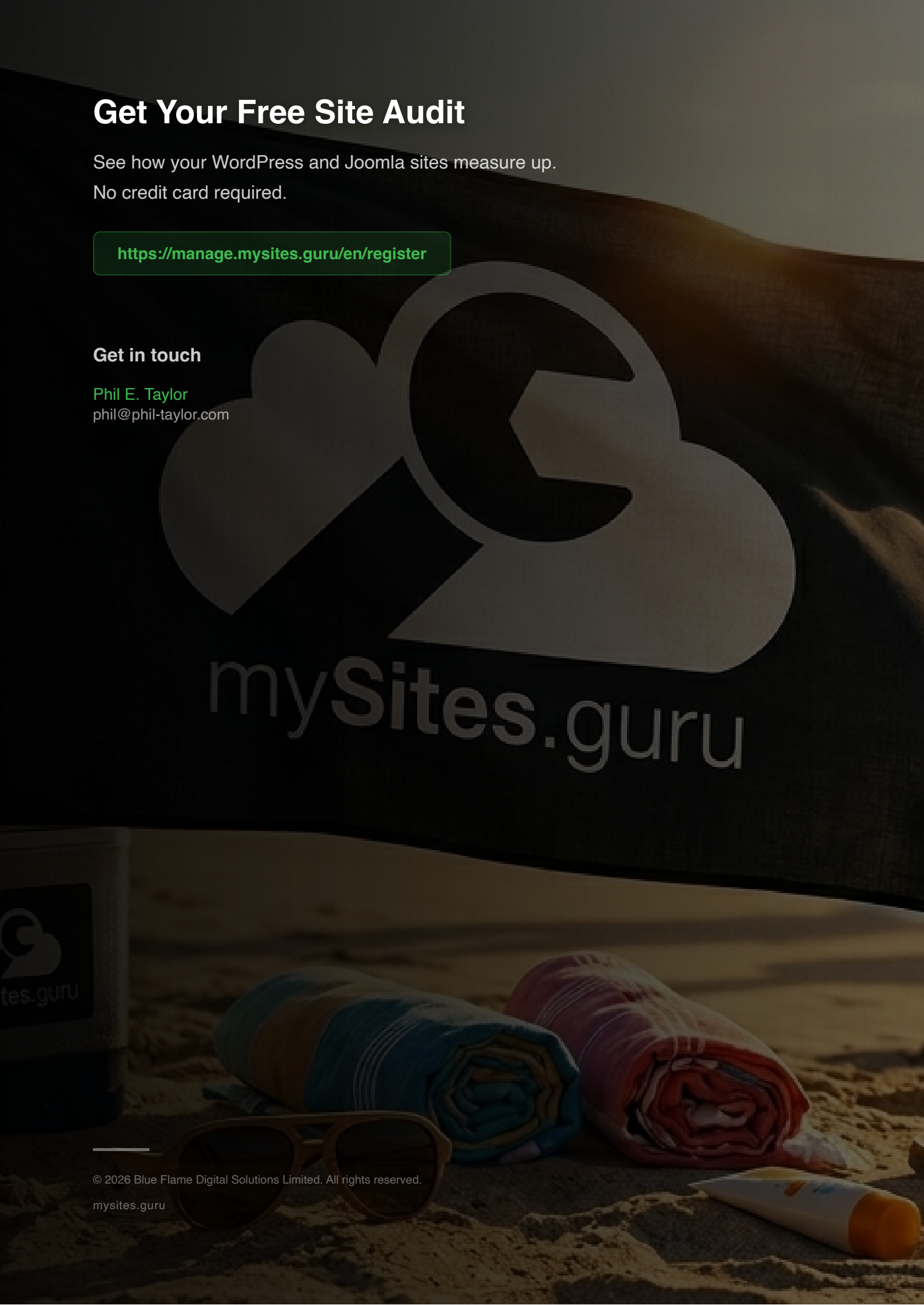
Treat the site as compromised. Don't just delete the file: the attacker got in somehow, so follow a full cleanup process or hand the site to fix.mysites.guru for a set-fee fix.

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