mySites.guru

Avada Builder Patches Two CVEs in 3.15.3

Avada Builder 3.15.3 patches an unauthenticated SQL injection and a Subscriber-level arbitrary file read across 1 million WordPress sites. Here's how to find every affected site you manage.

Phil E. Taylor | 12 May 2026

Avada Builder, the plugin bundled with the Avada theme by ThemeFusion, runs on around 1 million WordPress sites. On May 12, 2026, Wordfence published two CVEs and ThemeFusion shipped the final patch the same day.

The first is a Subscriber-level arbitrary file read that hands an attacker the contents of `wp-config.php`. The second is an unauthenticated time-based SQL injection that fires on any Avada site that ever had WooCommerce installed. Both are fixed in **3.15.3**. Update now.

If you manage WordPress sites for clients, the advisory is the easy bit. The work is figuring out which of your sites are still on 3.15.2 or earlier, then pushing the update to every one of them. Across the mySites.guru network we track Avada Builder on nearly 1,000 sites belonging to hundreds of agencies, so that work is real.

How do I find every Avada Builder site I manage?

Up to about ten sites, you can log in to each one and check the Plugins screen. Past that, you need a single view. The Plugins screen approach is slow, error-prone, and only tells you what was on disk the last time someone updated.

mySites.guru's WordPress Extensions inventory tracks every installed plugin across every connected site, refreshed on each snapshot. Search for "Avada Builder" (or the slug `fusion-builder`) and you get every version across your portfolio, grouped by version number, with a "Which sites?" button that lists exactly where each version is running.

mySites.guru subscribers: find every Avada Builder install across your sites

Open Avada Builder Extension Search

Lists every version of Avada Builder across your connected sites. Anything on 3.15.2 or earlier needs the 3.15.3 patch. Not a subscriber? Sign up free and connect your sites.

The same inventory feeds the vulnerability scanner. mySites.guru cross-references every installed plugin against the Wordfence Vulnerability API twice a day, so when a CVE like CVE-2026-4782 drops, sites running affected versions get flagged with a red warning banner. No security feeds to watch.

How do I bulk update Avada Builder across hundreds of sites?

Once you know which sites need updating, the mass updater handles the rollout. Tick the sites that need 3.15.3, push the update across all of them from one screen. The same workflow covers any plugin, theme, or core update, so the routine you wire up once works for every future CVE. Pair it with the Akeeba Backup or All-In-One Migration integration to back up each selected site first.

> **For agencies managing 50+ WordPress sites**
>
> Avada Builder ships with the Avada theme, so most clients running Avada have this plugin whether they know it or not. The patch is the easy bit; the inventory is the work. See how bulk updates work in mySites.guru.

If the patch needs more than a quick test on a particular site, disable Avada Builder until you can update. Deactivating it takes the vulnerable shortcode handler and AJAX endpoint out of play while you schedule a maintenance window.

Should I back up my WordPress sites before updating Avada Builder?

Yes, every time. Avada Builder uses its own page builder data format, and a bad update can leave you with broken layouts, missing elements, or corrupt page metadata. You often do not notice until a client phones to ask why the homepage carousel is empty.

Make sure each site has a recent backup before you push the update. A mySites.guru snapshot records config and version state, but it is not a restorable backup. For a

rollback you want the full one-click backup of files and database. The backup-all-sites workflow runs that backup across your whole portfolio in one click, so you can fire off backups for every Avada site, wait for them to finish, then run the update pass.

If you already run unlimited backup schedules, the patch goes on top of whatever you have scheduled, so the worst case is one extra restore point.

How do I watch for exploitation after patching?

The file read flaw has been at “partial fix” status since April 13. The full advisory dropped today, which is when most exploitation attempts start. Patching closes the door; monitoring tells you whether anyone walked through it before you got there.

mySites.guru’s real-time alerting covers the three signals that matter after a file-read or SQL injection CVE:

- File changes to core, theme, or plugin files. If someone grabbed `wp-config.php` salts and forged a session to drop a backdoor, the new file in `wp-content/` or a modified `wp-config.php` shows up before the next snapshot.
- New admin accounts. Anything created at administrator level fires an alert, whether it came in via the REST API, wp-admin, or directly through the database. Forging a session with leaked salts is one route to a rogue admin.
- wp-admin login activity, with IP, user agent, and timestamp. Unfamiliar logins on accounts that should not be active are the cheapest tell.

If any of those alerts fire, the malware scanner and the WordPress hacked recovery workflow handle the remediation side.

How do I show clients that their Avada site is now safe?

The patch is half the deliverable on a managed contract. The other half is the client knowing it happened. mySites.guru produces white-label client activity reports that list

every update applied, every backup taken, and every vulnerability cleared in a given window, branded with your own logo and domain.

After an Avada patch run, the same report tells the client which CVEs were closed, which sites went to 3.15.3, and that backups ran before each update. The custom client report builder lets you scope it to the work covered by their retainer rather than dumping the firehose of internal monitoring data on them.

What does CVE-2026-4782 actually do?

Detail	Value
CVE	CVE-2026-4782
CVSS	6.5 Medium
Type	Arbitrary File Read
Affected versions	All versions up to and including 3.15.2
Patched version	**3.15.3** (May 12, 2026)
Auth required	Yes - Subscriber or above
Bounty	$3,386.00
Researcher	Rafie Muhammad (via Wordfence Bug Bounty)

The flaw lives in Avada's `fusion_get_svg_from_file()` helper, reached from the `fusion_section_separator` shortcode via its `custom_svg` parameter. The function calls `$wp_filesystem->get_contents( $url )` with no checks on file type or path, then falls back to PHP's `file_get_contents()`. Nothing forces the input to be a `.svg`, so a `.php` file works just as well.

The shortcode is rendered via the AJAX endpoint `get_shortcode_render()` in the `Fusion_Builder_Front` class. That endpoint is nonce-protected, but the nonce is reachable by any logged-in user in vulnerable versions, and there is no capability check on the handler. A Subscriber account is enough.

This is the same pattern we covered in AJAX endpoints are a big CMS security blind spot: a developer protects the endpoint with a nonce, assumes that is enough, and forgets that the nonce is available to anyone with an account. Subscribers were never meant to invoke shortcode rendering. The check that would have stopped this is one `current_user_can()` call. The same root cause hit five other plugins in March 2026, including the AcyMailing flaw we wrote up where a registered Joomla user could escalate to admin through an AJAX endpoint with no capability check.

So any logged-in low-privilege user can call the shortcode with a path like `wp-config.php` and read the response. `wp-config.php` contains your database credentials, table prefix, and the eight keys and salts that secure WordPress sessions and password resets. Once those leak, the attacker can forge auth cookies, decrypt nonces, and pivot from "read a file" to "session as administrator."

If your Avada sites have open registration - membership sites, shops with customer accounts, anything where Subscriber accounts are created on signup - assume any Subscriber-level attacker has had this capability since Avada Builder 1.x.

Why an arbitrary file read of wp-config.php is so serious

A read primitive is not code execution, but it is one step away. The salts in `wp-config.php` are the inputs to WordPress's session signing. With those values an attacker can:

- forge `wp_session_tokens` cookies and walk into wp-admin as any user, including administrators
- read your database credentials and, if your DB is reachable from anywhere with those creds, dump the entire site contents
- decrypt any encrypted option stored in the database (some plugins encrypt API keys with the auth keys)

The fix in 3.15.3 restricts `fusion_get_svg_from_file()` to local SVG files and adds a capability check on the shortcode render endpoint.

What does CVE-2026-4798 actually do?

Detail	Value
CVE	CVE-2026-4798
CVSS	7.5 High
Type	Unauthenticated Time-Based Blind SQL Injection
Affected versions	All versions up to and including 3.15.1
Patched version	**3.15.2** (April 13, 2026)
Auth required	No
Bounty	$1,067.00
Researcher	Rafie Muhammad (via Wordfence Bug Bounty)

The post card items shortcode runs a SQL query against the `wc_order_product_lookup` and `wc_orders` tables to order results. The `product_order` parameter from `$_GET` is fed straight into the query's `ORDER BY` clause:

```
$args['order'] = ( isset( $_GET['product_order'] ) )
    ? sanitize_text_field( wp_unslash( $_GET['product_order'] ) )
    : $defaults['order'];

$query = "
    SELECT opl.product_id, MAX(opl.date_created) AS last_purchased
    FROM {$wpdb->prefix}wc_order_product_lookup AS opl
    INNER JOIN {$wpdb->prefix}wc_orders AS o
        ON opl.order_id = o.id
    WHERE o.status IN ('wc-completed', 'wc-processing')
        AND opl.product_id > 0
    GROUP BY opl.product_id
    ORDER BY last_purchased {$args['order']}
    LIMIT %d
";
```

`sanitize_text_field()` strips tags and HTML, but it does not escape SQL. The query passes through `$wpdb->prepare()` only for the `%d` placeholder; the interpolated `{$args['order']}` is appended as raw input. An attacker submits a payload like `last_purchased) CASE WHEN (...) THEN SLEEP(5) ELSE 0 END --` and reads response timing to extract data one bit at a time.

Why the SQL injection has a strange trigger condition

The query references `wc_order_product_lookup` and `wc_orders`, WooCommerce's HPOS (High-Performance Order Storage) tables. They only exist on sites that have run a recent WooCommerce.

But the tables persist after WooCommerce is deactivated or uninstalled. WooCommerce deliberately avoids dropping order data on uninstall, which is normally a feature, not a bug. That is what creates the trigger condition in the Wordfence advisory: "the vulnerability can only be exploited if WooCommerce was previously used and then deactivated."

That catches a surprising number of sites:

- Client stores that were set up, never used, and deactivated to "clean up"
- Sites that migrated off WooCommerce to a different store plugin
- Test sites that briefly enabled WooCommerce and forgot to drop the tables
- Sites running Avada demo content with the WooCommerce demo data still on disk

If WooCommerce has ever touched your database, you are in scope for this one, even if it is no longer active.

How to verify your sites are patched

Three checks, in order of speed:

1. **mySites.guru**: open the Avada Builder extension search and confirm every install is on 3.15.3 or later. Anything on 3.15.2 or earlier still needs updating.

2. **WP-CLI** on a single site: `wp plugin get fusion-builder --field=version`
3. **wp-admin**: Plugins page, find Avada Builder in the list, check the version.

If the version is 3.15.0, 3.15.1, or 3.15.2, update to 3.15.3. If WordPress reports no update available, check that the Avada theme is registered with a current ThemeFusion licence. Avada Builder updates come through ThemeFusion's update channel, not the WordPress.org directory, so a lapsed licence leaves the plugin pinned to an old version.

For sites you cannot patch right away, deactivating Avada Builder takes the vulnerable shortcode handler and AJAX endpoint out of play until you can schedule the update.

What to do if you suspect you were compromised

Both vulnerabilities have been public since Wordfence's advisory dropped today. CVE-2026-4798 has had a partial patch since April 13 in 3.15.2, so exploitation attempts may have been live for several weeks before the SQL injection patch dropped. The arbitrary file read in CVE-2026-4782 only got fully patched today.

If you find a vulnerable version and you also see signs of compromise (new admin accounts, unfamiliar files, mailer abuse), treat `wp-config.php` as leaked. Here is the checklist, with the mySites.guru tools that automate each step across your portfolio.

1. Rotate the database password and salts

The leaked `wp-config.php` contains your DB credentials and the eight keys and salts used to sign WordPress sessions. Both are now visible to anyone who exploited the file read.

```
wp config shuffle-salts
```

Rotate the database password in your hosting control panel and update `DB_PASSWORD` in `wp-config.php` to match. Salt rotation invalidates every existing session, so users

have to log in again. That is the point.

How mySites.guru helps: the universal user management view lists every user account across every connected site in one place, so once salts are rotated you can spot which accounts log back in and which never do.

2. Hunt for rogue admin accounts

If an attacker forged a session before you rotated salts, they could have created an administrator. Check the user table:

```
SELECT ID, user_login, user_registered, user_email
FROM wp_users
ORDER BY user_registered DESC
LIMIT 20;
```

Anything created since April 13, 2026 that you do not recognise is suspect.

How mySites.guru helps: universal user management sorts every account across every connected site by registration date. Filter for accounts created since 3.15.2 dropped and you see hundreds of sites in one screen, with no per-site wp-admin login.

3. Look for backdoors and modified files

A file read on its own does not plant code, but a forged admin session afterwards can. Look for unexpected PHP files in `wp-content/uploads/`, `wp-content/mu-plugins/`, and anywhere else uploads should not contain executable code.

```
find wp-content/uploads -name '*.php' -type f
find wp-content/mu-plugins -name '*.php' -mtime -30
```

How mySites.guru helps: the suspect content scanner runs the same pattern matching across every connected site. Anything it flags gets a one-click AI-powered

malware analysis that explains what the code does in plain English, so you can sort real threats from false positives without reading every flagged file by hand. The hidden files report flags orphan PHP files outside the normal core, theme, and plugin paths.

4. Confirm no core files were tampered with

If salts leaked and a forged admin session dropped a backdoor, the easiest hiding spot is a modified core file or theme file. Compare against the canonical hashes:

```
wp core verify-checksums
wp plugin verify-checksums --all
```

How mySites.guru helps: real-time file change alerts compare every PHP file's hash against the previous snapshot and email you when anything changes outside a normal update window. The deep security audit walks core, theme, and plugin files for tampered content as part of its run.

5. Hand it off if the site is compromised

If the site was actually exploited (extra admins, modified files, unfamiliar cron jobs) rather than just exposed, the cleanup is bigger than a plugin update. fix.mysites.guru handles the patch-plus-audit on a per-site basis: Avada Builder goes to 3.15.3, the site is checked for unfamiliar admin accounts and traces of file-read activity, credentials and salts are rotated, and the site comes back secure. Non-subscribers get a free month of mySites.guru included with the work.

Should I be worried about the next Avada CVE?

Avada Builder ships with the Avada theme, one of the most-installed paid themes on WordPress. A big install base attracts researchers. Two CVEs from one researcher in one disclosure round is not a fluke: the March 2026 four-plugin patch round hit

Elementor and Yoast SEO at similar scale, and the Essential Plugin supply-chain backdoor hit 31 plugins from a single acquired portfolio.

The answer is not "switch off Avada." It is the same answer as for every other widely-used plugin: keep the inventory current, patch within a defined window, monitor for changes, and have a rollback ready. mySites.guru's scheduled updates let you stage update windows for client sites so they do not all jump on day-zero releases. A 48 to 72 hour delay would not have helped here (the patch was the safe version), but it does catch supply-chain attacks like Smart Slider 3 Pro 3.5.1.35 where the malicious release was pulled within hours of release.

Agencies that handle CVEs cleanly are not faster readers. They have the inventory, the rollout, the monitoring, and the rollback already wired up before the CVE drops.

The disclosure timeline matters here

This one had a long tail:

- **March 21, 2026**: Rafie Muhammad submits both vulnerabilities through Wordfence Bug Bounty
- **March 24, 2026**: Wordfence validates the file read and discloses to ThemeFusion
- **March 25, 2026**: Wordfence validates the SQL injection and discloses; Wordfence Premium gets a firewall rule
- **April 13, 2026**: ThemeFusion ships 3.15.2 with the SQL injection patched (partial fix for the file read)
- **April 24, 2026**: Wordfence Free users get the firewall rule
- **May 12, 2026**: ThemeFusion ships 3.15.3 with the file read fully patched; Wordfence publishes the advisory

If you updated to 3.15.2 last month thinking you were safe, you closed the SQL injection. The file read was only partially mitigated. 3.15.3 is the version you actually need.

Further reading

- Wordfence advisory: 1,000,000 WordPress Sites Affected - full technical write-up with proof-of-concept and disclosure timeline
- CVE-2026-4782 on Wordfence Threat Intel - the arbitrary file read entry
- ThemeFusion (Avada) - the vendor. Check your licence is current to receive updates
- mySites.guru WordPress vulnerability alerting - how the automatic detection works
- How to update Joomla, Joomla extensions, WordPress, and WordPress plugins from mySites.guru - the bulk update workflow

Frequently Asked Questions

What is the Avada Builder CVE-2026-4782 vulnerability?

CVE-2026-4782 is an authenticated arbitrary file read in Avada Builder versions up to 3.15.2. A Subscriber-level user can call the fusion_section_separator shortcode with a malicious custom_svg parameter and read any file on the server, including wp-config.php with database credentials and cryptographic salts. The flaw is rated CVSS 6.5 Medium and fully patched in 3.15.3.

What is the Avada Builder CVE-2026-4798 SQL injection?

CVE-2026-4798 is an unauthenticated time-based blind SQL injection in Avada Builder versions up to 3.15.1. The product_order parameter on the post card items query is sanitised with sanitize_text_field() but not parameterised through wpdb prepare(). Rated CVSS 7.5 High and patched in 3.15.2, it only triggers if WooCommerce was previously installed on the site and then deactivated.

How do I find every WordPress site I manage that runs Avada Builder?

Manually you would log in to each site and check the Plugins page. With mySites.guru you open the WordPress Extensions inventory, filter for Avada Builder (slug fusion-builder), and click 'Which sites?' to list every connected site running the plugin along with its installed version. Hundreds of sites become a 5-second lookup.

Can I bulk update Avada Builder across all my WordPress sites?

Yes. mySites.guru's mass updater lets you tick a list of sites and push Avada Builder 3.15.3 to all of them from one dashboard. Pair it with the Akeeba Backup or All-In-One Migration integration to take a backup of each site first. The same workflow handles any plugin or core update across hundreds of sites without logging into each wp-admin.

Is the Avada Builder SQL injection exploitable on every WordPress site?

No. The SQL injection only fires when WordPress holds WooCommerce order lookup tables from a previously activated WooCommerce installation. Sites that have never run WooCommerce are not affected by CVE-2026-4798. The arbitrary file read (CVE-2026-4782) has no such condition and affects every Avada Builder install at 3.15.2 or earlier.

Does Avada Builder ship with the Avada theme?

Yes. Avada Builder (slug fusion-builder) is bundled as a required plugin with the Avada theme by ThemeFusion. Every Avada theme install also runs Avada Builder, which is why the

plugin sits on approximately 1 million WordPress sites. If you bought the Avada theme, you have the plugin too.

Does mySites.guru detect vulnerable Avada Builder versions automatically?

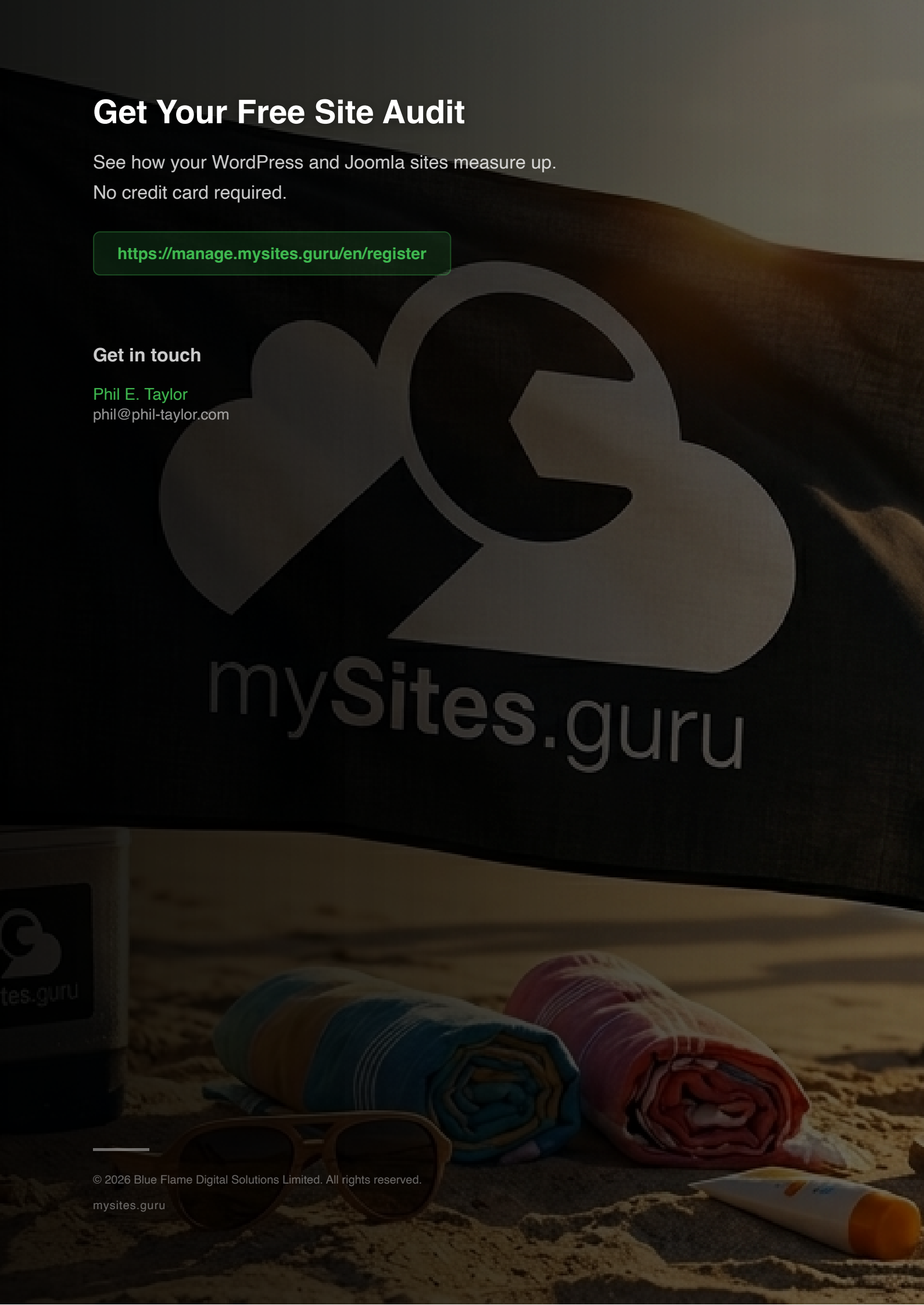
Yes. The vulnerability scanner cross-references every installed plugin against the Wordfence Vulnerability API twice daily. When you have Avada Builder 3.15.2 or earlier on a connected site, the dashboard flags it with a red 'vulnerable plugins' banner and links to the CVE details before you have to think about it.

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