



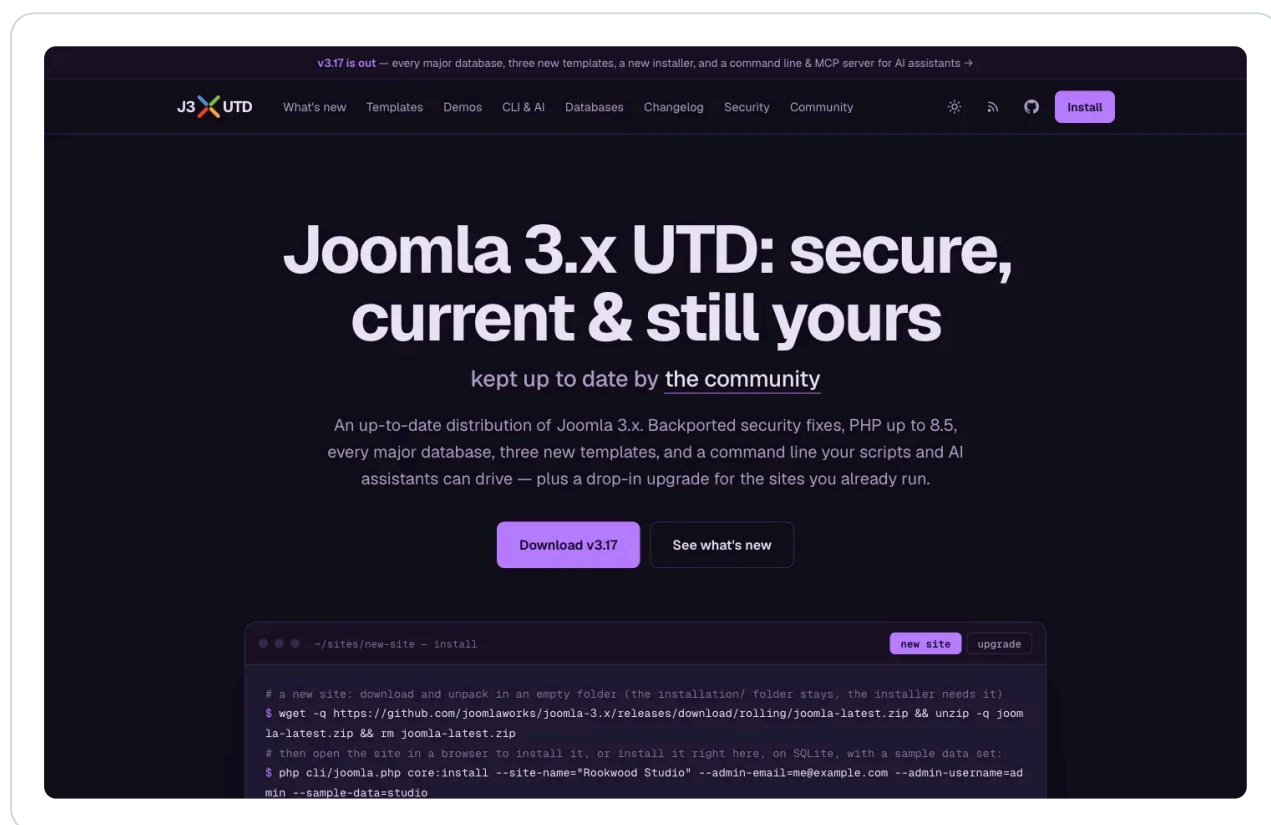
What is Joomla 3.x UTD (And why you don't need it)

Joomla 3.x UTD is a one-developer fork of Joomla 3 sold as secure. It has unsigned updates, its own security bugs and no way back to Joomla.

Phil E. Taylor | 12 October 2026

Joomla 3.x UTD describes itself as “an up-to-date distribution of the Joomla 3.x content management system, built to ensure code security, support modern PHP & MySQL/MariaDB versions & fix any broken behaviour that never got sorted before the release of newer major versions of Joomla.”

Its website goes further. Under the headline “secure, current & still yours”, it says UTD “keeps those sites secure and current” and “breathes new life into them”, which pitches it as the rescue for every Joomla 3 site left behind after support ended.



The Joomla 3.x UTD homepage on 12 October 2026.

The project behind that pitch is a hard fork of Joomla 3.10.20 eLTS, published by JoomlaWorks (the company behind K2) in April 2026 and written by one developer with an AI assistant. It ships at a pace that has already put its own security bugs onto live sites. Open Source Matters, Inc. and The Joomla! Project do not make it, endorse it or support it, as the project’s own footer says.

Joomla 3.x UTD is not an official Joomla product name or version.

It is a complete fork of end-of-life Joomla 3 code by a third party, JoomlaWorks Ltd. The Joomla project has never released a version 3.11 or later, and Open Source Matters and the Joomla project do not make, endorse or support it.

We covered UTD briefly in [Joomla 3 Didn't Fail. Your Retainer Did.](#), alongside the other projects trying to keep Joomla 3 alive. This post looks at it properly. Everything below comes from the public repository at github.com/joomlaworks/joomla-3.x, its website at j3xutd.joomlaworks.net, and JoomlaWorks' own blog. Each point links to its source.

But why this blog post?

Because people are confused. Most Joomla 3 site owners have no interest in the politics between a fork and the official project, or in whose policy says what. They want stability, and some assurance that their site will keep running safely. A shiny website promising Joomla 3 a long and secure future offers exactly that, and it is easy to believe. If that promise rests on one developer and falls through, the owners who trusted it are worse off than when they started: the site still needs migrating, and there is less time to do it.

People need clarity, so this post sets out who makes UTD and what it does and does not do, and links the source for each point so you can check it yourself.

Where did Joomla 3.x UTD come from?

Official Joomla 3 support ended on 17 August 2023. The paid Joomla 3 eLTS programme added 18 months of security fixes and ended on 17 February 2025.

The repository was created on 20 April 2026. Its first commits are "Initial upload for Joomla 3.10.20 eLTS", followed minutes later by "Initial upload of Joomla v3.11". Seven

releases followed in under six months, numbered 3.11 to 3.17, the latest dated 10 October 2026.

"Secure" is the headline, so start there

"Secure" is the first word of UTD's headline, and security is the reason anyone would run a Joomla 3 fork instead of migrating. So that is where it has to be judged, and it is where it is weakest.

Look at the chain that keeps a CMS site safe: who writes and reviews the fixes, how a vulnerability gets reported, how updates reach the site, and whether they can be checked when they get there. Official Joomla has a team at each step. Reports go privately to the Joomla Security Strike Team (JSST), who confirm them, fix them with the core maintainers, and publish an advisory with the release. I have reported Joomla core vulnerabilities through that process myself. Update packages come with checksums, and since Joomla 5.1 they are signed with TUF.

UTD has one developer and an AI assistant at every step. No one reviews the fixes. There is no private way to report a vulnerability. Updates are an unsigned zip, rebuilt on every commit and published with no checksum. Its own changes have already added a security hole, and the servers it tells owners are fine are years past their own security support. The sections below take these one at a time.

1

Developer

Every one of 158 commits

0

Checksums in the update feed

Official Joomla publishes three

2

Little WAF filters

None for SP Page Builder

10

Rebuilds of the stable zip

Since 10 October 2026

Read from the [joomlaworks/joomla-3.x](https://github.com/joomla-works/joomla-3.x) repository and its update feed on 12 October 2026.

The core is not what is getting Joomla 3 sites hacked

Even a perfect core fork would miss where the damage comes from. In [Joomla 3 Didn't Fail. Your Retainer Did.](#) we measured the Joomla 3 sites connected to mySites.guru on 28 August 2026: 4.90% were flagged as hacked against 0.98% on Joomla 6, and 22.3% were running at least one extension with a known published vulnerability. The most common were YOOtheme Installer, Regular Labs' Cache Cleaner and Advanced Modules, SP Page Builder and JCE, and for six of the eight most common, the 2026 fix cannot be applied on Joomla 3 at all.

UTD's answer to the extension problem is the Little WAF, a plugin added in 3.16 that its website says "filters known attack signatures against abandoned or historically vulnerable third-party extensions". That sounds broad. The code in `plugins/system/littlewaf/littlewaf.php` has exactly two filters, and it has had only those two since it first shipped: one blocks Regular Labs Sourcerer's `{source}` tag in requests, the other blocks Modules Anywhere's `{module}` and `{modulepos}` tags. Both target real problems, and [Sourcerer's CVE-2026-74253](#) was being exploited when the filter was written. The plugin is also disabled by default, and its own description says it is "not a substitute for updating or removing the vulnerable extension itself".

That leaves everything else in the table above. There is no filter for SP Page Builder, including the `uploadCustomIcon` file upload flaw (CVE-2026-48908) that was exploited in the wild, or the unauthenticated SQL injection and file inclusion holes that followed it. I reported those to JoomShaper myself, 14 SP Page Builder vulnerabilities in all this year, and SP Page Builder 6.8.0 and later will not install on Joomla 3. There is nothing for YOOtheme or JCE either. A site owner who reads "secure, current & still yours" and moves a Joomla 3 site onto UTD keeps every one of those extensions, gets two tag filters if they think to switch the plugin on, and now has a core that tools cannot identify as well.

A one-man band shipping its own security bugs

The 3.17 release notes list "22 security fixes" in fourteen entries. Nine are marked "(stock Joomla 3)". One is marked "(since 3.12)": the `{loadmoduleid}` tag "showed

any module to anyone, whatever its access level or publishing dates". The CHANGELOG explains that the 3.12 change letting it show modules not assigned to the current menu item "dropped those checks too", so an author could display any module, including private ones, inside an article. The fork introduced that access-control hole on 21 May 2026 and fixed it on 10 October, and then counted the fix in its security total.

Other bugs were also the fork's own work. Every Help button in the administrator showed "not found" from 3.11 to 3.17, because the new version numbers point at help screens that do not exist on joomla.org. The Little WAF plugin added in 3.16 showed no version, date or author on upgraded sites, and was fixed a week later in 3.17. The issue tracker has "SEF canonical tags gone after J 3.14 update" (#16), "redirect after updating to 3.13" (#6), "install 3.15 when banners component has been removed results in an error" (#18) and "Beez3 not fully removed" (#7).

The security page has the same problem. Most of the CVE IDs it lists resolve on cve.org, but "CVE-2026-352212" (described as a com_tags SQL injection) does not exist. If an ID on a security list does not resolve, you cannot check that the fix it describes is the fix you think you have.

Seven releases in under six months, two of them a week apart, with breaking changes in minor versions, regressions reported after updates and a security hole of the fork's own making. That is sloppy release engineering, and it comes from a single maintainer: every commit is from the same account, and no change has gone through a reviewed pull request. Joomla 3 sites need careful maintenance far more than they need a new installer and an MCP server.

Security fixes no one else has checked

UTD says it backports Joomla core security fixes from Joomla 5 and 6, and also fixes "issues listed upstream as affecting only Joomla 4.0 and later". For one template manager XSS it says it fixed "three code paths" where "the official fix covered one". It may be right, but no one outside the project has checked. When the JSST fixes a core vulnerability, the fix is reviewed by more than one person and released alongside a public advisory. A UTD backport is one developer's port, with an AI assistant, of a patch written for a different codebase, pushed straight to main and then to every site that

updates. The changelog shows how that can go wrong: the `{loadmoduleid}` hole above was introduced by a UTD change. When we backport a Joomla 3 fix, we publish how we verified it against 3.10.12 so anyone can check our work.

Backports arrive weeks after the attackers know

A backport can only start once the fix is public. UTD is not part of the Joomla security process, so it gets no advance notice of a vulnerability: it sees the fix when Joomla publishes the patched release and its advisory, at the same moment as everyone else, including the people who read advisories to find sites to attack. A site on official Joomla 5 or 6 can update that day. A UTD site waits for one developer to port the fix to a different codebase and ship a release.

UTD's own security list shows how long that takes. Joomla's 26 May 2026 security release fixed a local file inclusion in the `HTMLView` layout parameter (CVE-2026-40383), an XSS in content history (CVE-2026-30894) and a transport encryption downgrade (CVE-2026-48902). UTD shipped those fixes in 3.15 on 18 July, 53 days later. In between it released 3.13 on 31 May and 3.14 on 4 July, and neither included them: 3.14 was PHP 8.5 deprecation fixes and language file versioning. The fixes from Joomla's 7 July release followed in 3.15 eleven days after they were published, and CVE-2026-21631, published on 1 April, waited until 3.12 on 21 May. For seven weeks in the middle of 2026, a fully updated UTD site was missing a published file inclusion fix that every updated Joomla 5 and 6 site already had.

No tests and no build tooling

The official Joomla repository comes with the scaffolding that lets anyone check a change before it ships. The Joomla 3.10.12 tag has a `tests/` folder with unit, Codeception and JavaScript tests, a `phpunit.xml.dist` to run them, Drone CI configuration that runs them on every pull request, and a `build/` folder with the release script, the deleted-file check and the coding standard. It also has a `composer.json` and `composer.lock`, which record exactly which versions of the bundled third-party libraries a release uses and let a maintainer update them.

The UTD repository has none of it. There is no `tests/` folder, no PHPUnit configuration, no CI beyond the workflow that zips the main branch, no `build/` folder and no root `composer.json`. It still ships the 580 files of `libraries/vendor` that Joomla 3 bundled, but without the Composer files no one can rebuild or update those libraries in a reproducible way. Every change, including every security backport, goes from one developer's editor to every site that updates without a single automated test running in between.

Updates no one can verify

Sites get UTD updates from `joomlaworks.github.io/joomla-3.x/extension.xml`. It offers "Joomla! 3.17.0", tagged `stable`, and the download is `releases/download/rolling/joomla-latest.zip`. The project's own workflow, `.github/workflows/rolling-release.yml`, deletes and rebuilds that zip "on every push to main". Version 3.17.0 is therefore whatever the main branch looked like at the last commit, and it has been rebuilt ten times since 10 October. There is no release branch, no release candidate and no test stage between a commit and your live site. You cannot download an earlier version to roll back to either: the v3.15 release on GitHub has no package attached.

Those packages also cannot be checked. Joomla's own update feed publishes a SHA-256, SHA-384 and SHA-512 checksum for every package, and UTD still has the code that checks them (`isChecksumValid()` in `com_joomlaupdate/models/default.php`), but its feed gives no checksums to check against. It could not, because the zip changes with every commit. Since Joomla 5.1, official core updates are also signed with TUF (The Update Framework). A UTD site installs whatever unsigned zip one GitHub account last produced, so anyone who gets into that account can push code to every UTD site that updates.

"Secure", on servers that are not

The README says sites on PHP 7.1 to 7.3 "will still be offered updates to this distribution", with PHP 7.4 as a recommendation rather than a requirement. PHP 7.1, 7.2 and 7.3 have had no security support for years, which is why we flag the PHP

version on every site. The minimum databases are MySQL 5.5.3, MariaDB 5.5 and SQL Server 2008 R2, all long out of support. Telling owners that a site on an unpatched server is “secure” because the CMS code is patched gets the risk the wrong way round. UTD’s core audience is already there: JoomlaWorks’ own K2 telemetry puts 92.7% of active K2 sites on PHP older than 8.2.

It can hide sites from vulnerability scanners

Vulnerability rules for Joomla core, including the ones mySites.guru checks every site against, are written against Joomla’s own version ranges, for example “3.0.0 to 3.10.x”. A UTD site reports “Joomla! 3.17.0”, which sits outside every one of those ranges. A scanner will report it as unaffected or as unknown, whether or not the fix is actually in its code, and UTD’s own changelog admits the numbering “would also ‘trip’ some scanners”. We see this from the inside, because mySites.guru is an inventory tool: a site reporting a version that never existed is a site we cannot tell you is safe or unsafe. We have already watched the same failure on a smaller scale, when JoomShaper reused one version string across two different security releases and no inventory tool could tell a patched site from an unpatched one. UTD has that problem for the whole core, permanently and by design. A distribution sold on security should not make its sites harder to audit.

No way to report a vulnerability

The repository has no **SECURITY.md**, no contributing guide and no published way to report a vulnerability privately. GitHub’s community health check scores it 50%. If someone finds a Joomla 3 core vulnerability tomorrow, there is no confidential route to the one person who can fix it, and the public issue tracker is not the place for one.

Three new vulnerabilities of UTD’s own

While researching this post, we identified three security issues in Joomla 3.x UTD that do not exist in the official Joomla 3.10.20 eLTS release it was forked from. They were introduced by UTD’s own changes. We will disclose them responsibly to JoomlaWorks shortly and publish the details once fixes are available. With no **SECURITY.md** and no

private reporting route, we will have to find our own way to reach the one person who can fix them.

Joomla 3.x UTD's bundled libraries are as old as Joomla 3

UTD calls itself an up-to-date distribution, so we checked every third-party library it bundles against the [OSV vulnerability database](#) on 12 October 2026. Not one of the 34 PHP libraries in `libraries/vendor` has been updated: UTD's `installed.json` lists exactly the versions official Joomla 3.10.12 shipped in July 2023. The JavaScript libraries are the same ones too. Official Joomla 3 hand-patched some of them in place, and those patches are still there (jQuery 1.12.4, for example, keeps Joomla's backported prototype pollution and `htmlPrefilter` fixes), so we have left anything already patched out of this list.

Library	Version UTD ships	Known issues not fixed in that version	Fixed in
TinyMCE (the "legacy" editor existing sites keep)	4.5.12, built July 2020	17 advisories published since that build, including three High-severity XSS from June 2026 (CVE-2026-47759 , CVE-2026-47761 , CVE-2026-47762)	Only TinyMCE 5 to 8; the 2026 three in 7.9.3 and 8.5.1
Bootstrap (JavaScript used across the administrator)	2.3.2	Six XSS advisories, including CVE-2018-14040	3.4.0 or 4.x, and no release at all for CVE- 2024-6485

Library	Version UTD ships	Known issues not fixed in that version	Fixed in
		and CVE-2024-6485	
MooTools	1.4.5	Prototype pollution (CVE-2021-23432) and a High-severity ReDoS (CVE-2021-32821)	No fixed version: the project is abandoned
symfony/yaml	2.8.52	Three parser denial-of-service advisories from May 2026 (CVE-2026-45304 , CVE-2026-45305 , CVE-2026-45133), Low	5.4.52 and later only
paragonie/sodium_compat	1.17.1	CVE-2025-69277 , Moderate	1.24.0

None of these is rated Critical, and whether each one can be reached depends on how Joomla uses the library. Most TinyMCE and Bootstrap issues need someone who can already edit content or load a crafted page. But these are the same libraries official Joomla 3 was shipping when it reached end of life, and UTD's own release notes keep TinyMCE 4.5.12 as the editor for every existing site "until they switch in Global Configuration". A distribution that says it ensures "code security" and has fixed "every PHP deprecation we could find" has not updated a single bundled PHP library, and with no `composer.json` in the repository it has no straightforward way to.

A hand-patched fork of jQuery from 2016

The jQuery that UTD loads in the administrator, and on any site whose template or extensions ask for it, is not a jQuery release at all. The first line of `media/jui/js/jquery.js` reads "jQuery JavaScript Library v1.12.4-joomla". It is jQuery 1.12.4, released on 20 May 2016, with security fixes the Joomla project patched in by hand over the following years: one for a jQuery XSS issue (CMS issue 19464), a prototype pollution guard taken from a [jQuery pull request](#), and a replacement `htmlPrefilter()` for [CVE-2020-11022](#) and [CVE-2020-11023](#), adapted from a Drupal backport. Each change is marked "ADDED by Joomla" or "Modified by Joomla" in the file. jQuery itself is now on [4.0.0, released on 18 January 2026](#), and the jQuery Migrate 1.4.1 bundled with it dates from the same day in 2016.

We compared the files with official Joomla 3.10.12, and `jquery.js`, `jquery.min.js`, `jquery-migrate.min.js`, `bootstrap.min.js` and `mootools-core.js` are byte-for-byte identical. UTD is shipping a fork of jQuery inside its fork of Joomla, and has not touched it. No one upstream maintains those Joomla patches any more, because the Joomla project stopped when Joomla 3 reached end of life, and nothing in UTD's repository rebuilds or tests them. The library still reports itself to the browser as `jQuery.fn.jquery = "1.12.4"`, so scanners that read the version flag the site as running a vulnerable jQuery 1.12.4, and no one can tell which of the later jQuery fixes are in this copy without reading it line by line.

UTD's three new templates load no jQuery, Bootstrap or MooTools at all, which is a sensible choice. The administrator, Protostar, every older template and most Joomla 3 extensions still do.

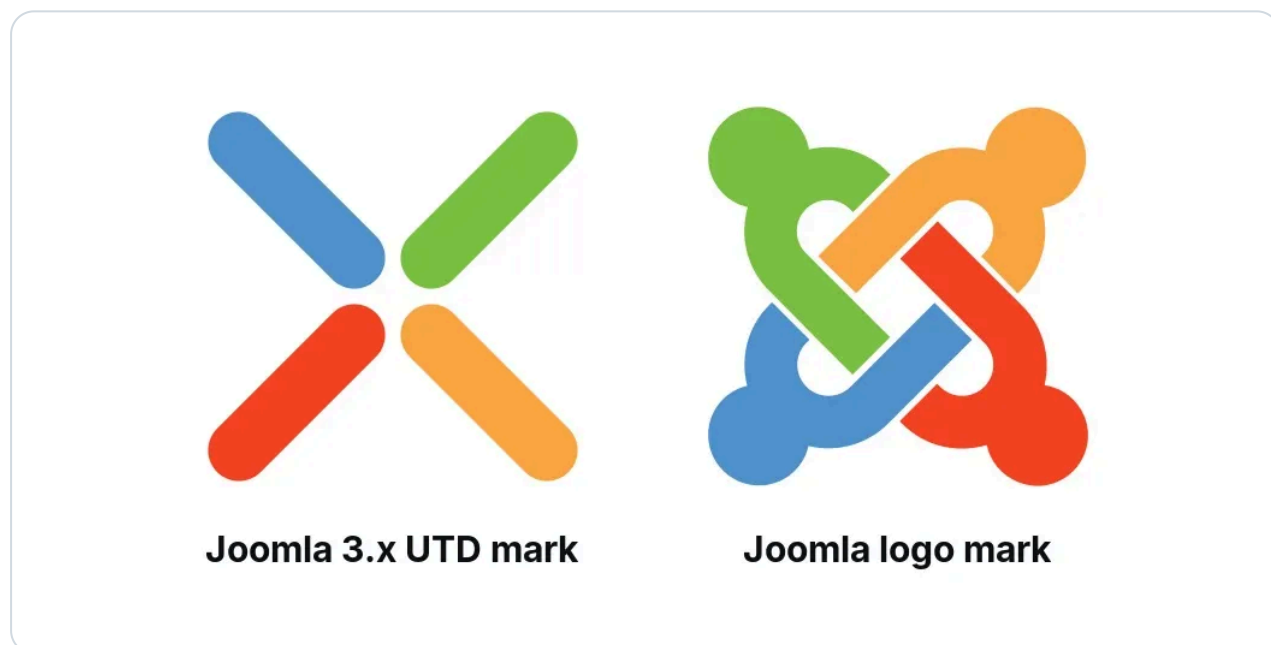
To be fair to UTD, its new TinyMCE 8 editor is Tiny's current release, 8.9.2 from 23 September 2026. Even that bundles DOMPurify 3.4.11, which has three advisories fixed in 3.4.12 and later, although that is a choice Tiny made, not UTD.

Why Joomla 3.x UTD confuses people

It puts the Joomla name and logo on a fork

The product name starts with "Joomla". The footer says "This project does not use the Joomla! logo", yet the fork's own rewritten installer and administrator login page both render it: `installation/template/index.php` line 60 and `administrator/templates/isis/login.php` line 76 load `joomla-logo.svg` inline, with a new code comment reading "The Joomla! logo inline". Both pages were redesigned for this fork, so the logo is there by choice.

UTD's own logo borrows from Joomla's too. Its mark, `assets/logo/j3xutd-mark.svg`, is four rounded strokes arranged as a cross in exactly Joomla's four brand colours, `#5091cd` blue, `#7ac143` green, `#f9a541` orange and `#f44321` red, set at the same diagonal as the four interlocking arms of the Joomla logo. It is not a copy, but next to "Joomla 3.x" in the product name, it is built to be read as part of the Joomla family.



UTD's mark (left) uses the same four colours as the Joomla logo (right).

The logo file has one more detail. It contains embedded C2PA content credentials, signed with Anthropic's "Claude Content Signing" certificate, that record the action `com.anthropic.claude.provided` and the description "Claude provided this file at the request of a user and may have created or modified the file contents." So the logo of a project that calls itself "kept up to date by the community" came out of an AI assistant, like 33 of its commits.

The Open Source Matters [trademark FAQ](#) is clear about forks. Asked whether you can take the Joomla codebase and rename it, it answers: "Yes, so long as you abide by the terms of the GNU GPL and if the new name and logo do not violate the trademarks." The GPL covers the code, and the FAQ says the licence is "designed for software not for logos or names". So the expectation is that a fork gets its own name. OSM also says it would "prefer that names of extensions be of the form [Extension name] for Joomla! Rather than Joomla! [Extension name]", and "Joomla 3.x UTD" is exactly the second form, on a whole CMS rather than an extension.

Put the pieces side by side: "Joomla" as the first word of the product name, version numbers that continue Joomla's own 3.x line, a **PRODUCT** constant that still reports "Joomla!", and the Joomla logo on the installer and login screens. Each one on its own could be explained away. Together they look like a fork presenting itself as Joomla, which is the confusion trade marks exist to prevent. The FAQ says as much: "The trademark also prevents confusion of other organizations or companies with Joomla! or Open Source Matters."

The JoomlaWorks website footer says its use of the Joomla name and logo "is permitted under a limited license", which is normal for a long-standing extension vendor. I can find no public statement that the licence covers naming a fork of the CMS itself "Joomla 3.x UTD". If it does, JoomlaWorks should say so. If it does not, this is a question for Open Source Matters.

The version numbers look like official Joomla releases

Joomla never released a 3.11. The Joomla 3 line stopped at 3.10.x, and so did eLTS. UTD starts at 3.11 and is now at 3.17, so a site running it reports a Joomla version the Joomla project did not ship.

It goes further than the label. In [libraries/src/Version.php](#) the **PRODUCT** constant is still **'Joomla!'** and the copyright line is still Open Source Matters, so extensions, update servers and security scanners see "Joomla! 3.17.0". The 3.14 changelog even notes that the new language file versioning "would also 'trip' some scanners".

The numbering also breaks semantic versioning. Minor releases have made breaking changes: 3.12 removed the Beez3 and Hathor templates, and 3.16 removed four cache and session drivers and raised the minimum database versions. Under semver, changes like that need a major version. The site also describes “one rolling release”, so the version number does not tell you much about what code you are running.

Even its own developer cannot settle on a name

Is it Joomla 3.x UTD, J3xUTD or just Joomla 3.17? It depends where you look, and every one of these comes from JoomlaWorks itself:

Where	What it is called
The website headline, the README and the administrator status bar	“Joomla 3.x UTD”
The logo and the domain	“J3”, the four-colour cross and “UTD”, read as “J3xUTD” (the logo file is labelled “J3xUTD mark”), at j3xutd.joomlaworks.net
The April 2026 announcement	“an open distribution of Joomla 3.x (we call it Joomla 3.x UTD for now)”
The update feed’s own source	“Joomla 3.x (JoomlaWorks Security Distribution)”
The update Joomla Update offers you	“Joomla! 3.17” and “Joomla! 3.17 CMS”
The GitHub releases	“Joomla v3.11”, “Joomla v3.14”, “Joomla v3.15” and “Rolling Release (v)”
The GitHub repository	<code>joomla-3.x</code> , “Joomla 3.x development continued to ensure code security and support modern PHP versions”

Six months in, the name is still officially "for now", the acronym is spelled out only once, at the top of the README, as "(up-to-date)", and the two places a site owner is most likely to see it, the update screen and the release list, do not use the UTD name at all. They call it Joomla.

"Kept up to date by the community"

The website's hero says the project is "kept up to date by the community", and the footer calls it "an independent community project". The repository shows something different:

- All 158 commits come from one GitHub account, Fotis Evangelou's (**fevangelou**).
- 33 of those commits have a **Co-Authored-By: Claude Opus 5.5** trailer, all from October 2026.
- The only pull request from anyone else (#13) was closed without being merged.
- There are 14 issues from 5 people, and 8 discussions, 6 of them started by one user.

A small group reports bugs and Fotis writes the code, with an AI assistant. That is a legitimate way to run a project, but it is a one-person project and a single point of failure.

Its help system is the official Joomla one

UTD has no documentation site of its own. Its website has no documentation section, and the **docs/** folder in its repository holds only the update feed and sample data. Everything it has added, from the new installer and the 100-command CLI to the MCP server, SQLite and the Little WAF, is documented in the README and one AI-GUIDE.md file.

Press a Help button in a UTD administrator and you go to the official Joomla project. The CHANGELOG explains that from 3.11 to 3.17 those buttons showed "not found", because the Joomla help site only has screens up to 3.10, and that "Versions 3.11 and later now use the 3.10 help screens". So a UTD user who needs help is sent to Joomla's

own 3.10 help screens, run by the volunteers of the project that does not support UTD, to read documentation written for a version they are not running. The same goes for the Joomla forum, Joomla Stack Exchange and every Joomla 3 tutorial on the web: all of it describes official Joomla 3, and none of it knows what UTD changed.

Borrowed goodwill and selective statistics

Forking is fine. Forking well means taking the code and leaving the name behind, and open source has done it that way for decades. Joomla itself began in 2005 as a fork of Mambo, and the first thing the developers did was choose a new name. LibreOffice left OpenOffice's name with Oracle, MariaDB left MySQL's, and when WordPress 5.0 brought in the block editor, the people who wanted to keep WordPress 4.9 alive called their fork ClassicPress. Each project had to earn its own reputation instead of borrowing its parent's. JoomlaWorks knew this in 2024, when its own plan for the fork started with a rebrand.

Keeping the name lets UTD trade on twenty years of goodwill it did not build. The Joomla name means something to site owners because of the volunteers who wrote the code, triaged the security reports, translated it into dozens of languages, ran the events and answered forum questions for two decades. UTD arrives as "Joomla 3.x", version 3.17, with the Joomla logo on its login page, and borrows that trust on day one. The trust belongs to the Joomla project and the people who earned it.

The statistics are borrowed the same way, and chosen to suit. In May 2024 JoomlaWorks used W3Techs to argue that Joomla 4 and 5 had failed: "They account for around 0.3% of ALL Joomla sites according to W3Techs", so "adoption for these versions is close to non-existent". It set aside Joomla's own figures, which it said "still show around 10% of sites use Joomla 4/5". In October 2026 the v3.17 announcement goes back to W3Techs for the headline that Joomla 3 runs on "more than 50% of all installed Joomla sites". It does not mention that the same source now puts Joomla 4, 5 and 6 at 37.9% of Joomla sites, with Joomla 5 alone at 20.5%. Quoting a source when it helps and leaving out the rest of the same page is marketing, not evidence.

A dead end on the way back to official Joomla

UTD rewires Joomla Update. In `com_joomlaupdate/models/default.php`, the Default, Testing and "Next" channels all point at JoomlaWorks' feed. The code describes "Next" as "Minor & Patch Release for Current version AND Next Major Release", and it is the channel a Joomla 3.10 site uses for the one-click upgrade to Joomla 4. Switching to a custom URL does not help: Joomla's official Joomla 4 upgrade feed (`update.joomla.org/core/j4/default.xml`) targets sites on 3.10, and a UTD site reports 3.17. Once a site moves to UTD, the official upgrade route is gone. Even on official Joomla, switching update channels can fling a site onto the wrong release, and UTD has rewired every one of them.

The database choices make it worse. UTD's headline 3.17 feature is "every major database", including SQLite and SQL Server. Joomla 4, 5 and 6 support only MySQL, MariaDB and PostgreSQL, and Joomla 4 dropped SQL Server. A site built on UTD's SQLite or SQL Server support has no official Joomla to move to. A SQLite site can only get out through UTD's own `database:convert --to=mysql` command, run before any migration, on a driver the project itself labels experimental. A SQL Server site cannot even do that: the README says `database:convert` and the other database commands "don't support SQL Server". A site that could have upgraded itself from 3.10 now depends on one maintainer's tools to leave his fork.

Your Joomla 3 extensions are not guaranteed to keep working

Joomla 3 extensions may install and run on UTD today. The project does not promise they will keep doing so, and the published roadmap points the other way.

The UTD README describes "a new fork on the way" that is "a stripped down version of Joomla 3.x with all non-essential extensions removed", where "com_content (and anything related) is removed entirely" and replaced by K2. The v3.17 announcement calls that fork "a separate project" that is "moving along very actively", and the README calls UTD's CLI and MCP server "the seed" for it. So the same single maintainer is splitting his time between UTD and a successor that drops the component most extensions are built on. Most third-party Joomla 3 extensions depend on com_content:

content plugins, article modules, custom fields, SEF routers and page builders. The compatibility promise in that same sentence is “true backwards compatibility with past releases (of the fork)”, meaning with UTD itself, not with the Joomla 3 extensions you run.

The minor releases so far have already removed core templates and drivers. The README also asks extension developers to change their update XML so it does not stop at Joomla 3.10.x, which means support depends on third-party developers who have made no commitment to this fork. The Joomla Extensions Directory feed itself stops at 3.10.

So far, no other developer has taken up that request. When we searched in October 2026, the only extensions declaring support for Joomla 3.x UTD were JoomlaWorks’ own: K2, whose [JED listing](#) says it is “committed to be 100% compatible” with the distribution, and the paid Simple Image Gallery Pro, [released as “PHP 8.x compatible for Joomla 3.x UTD”](#). We could not find a single third-party developer announcing, testing or releasing an extension for it, and a GitHub code search for “Joomla 3.x UTD” outside JoomlaWorks’ own repositories found nothing but our own pages. An ecosystem of one vendor is not much of an ecosystem, and one of UTD’s most active users said as much in its own discussions, under the title “J3 development is good but we miss an ecosystem”.

Joomla developers would have to support a second, older framework

Building for UTD means building for Joomla 3, which is a different platform from Joomla 4, 5 and 6. Joomla 3 extensions use the legacy MVC classes (`JControllerLegacy` , `JModelLegacy`), Bootstrap 2, jQuery and MooTools, and no namespaced extension structure. Since Joomla 4, extensions use namespaced MVC, a dependency injection container with `services/provider.php` , the Web Asset Manager and Bootstrap 5. Code written for one does not run on the other without real porting work.

A developer who wants to support UTD users therefore keeps a second codebase for an old framework that only one maintainer is moving forward. If the roadmap’s K2-

based fork ships, extensions that rely on com_content would need yet another target. The README's request that they extend their Joomla 3 update XML asks them to take on that work for free.

What is Joomla 3.x UTD really for?

JoomlaWorks has said why in its own words, and its [Joomla 3 record](#) is on our Joomla 3 security hub. In a blog post on 30 May 2024, "[K2 will not be made available for Joomla 4/5 - change of course](#)", the company called Joomla 4 and 5 "failed CMS releases", said "we don't plan on giving up on K2 (or Joomla 3)", and said its "goal is to pivot and work on a possible fork". The first phase of that plan was rebranding, so that existing sites could move their content from com_content to K2.

UTD and its planned leaner successor are that plan, except that UTD skipped the rebranding and kept the Joomla name. UTD keeps K2's natural home alive, and the successor replaces Joomla's own content component with K2. It also keeps a market alive for JoomlaWorks' paid products: the blog sidebar advertises Simple Image Gallery Pro as "PHP 8.x compatible for Joomla 3.x UTD", beside the "Commercial Templates" menu. And it pushes hard into AI, with a built-in MCP server and a CLI that the README says brings it "closer to becoming a truly agentic CMS".

Who it is for: the sites its own developer runs

UTD's developer owns a second business that shows who the fork is for. [Nuevvo](#) is Fotis Evangelou's Athens web development and managed hosting company, listed alongside JoomlaWorks and K2 on [his own site](#) and his GitHub profile, and its homepage lists 42 active clients, most of them Greek news and magazine sites. We loaded each client's homepage on 12 October 2026. Twelve still run K2 on Joomla, with K2's own asset paths all over the page: [dnews.gr](#), [reporter.gr](#), [ilialive.gr](#), [news4health.gr](#), [advertising.gr](#), [allstarbasket.gr](#), [healthstat.gr](#), [ienergeia.gr](#), [bikeit.gr](#), [byblacks.com](#), [karditsalive.net](#) and [gloman.gr](#).

Those twelve are UTD's natural first customers, sites with years of articles in K2 and no K2 for Joomla 4 or later, and keeping them on a maintained Joomla 3 is a reasonable

thing for their developer to want. It is a narrower purpose than “secure, current & still yours” for every Joomla 3 site. The rest of the same client list points the other way: 20 of the 42 now run WordPress, and the Greek broadcaster SKAI’s site, skai.gr, runs Drupal 10.

None of that is wrong in itself. Joomla is GPL software and anyone may fork it.

Site owners deciding where to take a Joomla 3 site should understand what they are choosing.

One company’s fork, maintained by one developer, built around that company’s own extension, that presents itself to your software as an official Joomla version the Joomla project did not release.

Joomla 6 or Joomla 3.x UTD?

For almost everyone, Joomla 6, and its technical requirements are modest. It is the official release, its vulnerabilities go to a security team rather than one person, its updates are checksummed and signed, and it is where extension developers are building today. A Joomla 3 site moving to UTD still faces a migration later, just a later and possibly harder one, because the K2-based successor drops com_content.

JoomlaWorks pitches UTD with the W3Techs figure that Joomla 3 runs on more than half of all Joomla sites. That number is right, even if the rest of the page it comes from is left out: W3Techs puts Joomla 3 at 51.7% of Joomla sites on 12 October 2026, against 37.9% for Joomla 4, 5 and 6 combined. What it does not show is a market waiting for a fork. W3Techs counts sites that still identify as Joomla 3, and a site that has had no update since support ended in 2023 is mostly a site no one is looking after. Our own figures point the same way, and they come from the well-looked-after end of the market. Of the Joomla 3 sites connected to mySites.guru in August 2026, sites whose owners pay every month for monitoring, 68.2% had not even updated to 3.10.12, the final Joomla 3 release. If owners who are paying for monitoring did not take a free point release from the official project, the owners of unmonitored sites are

not going to go looking for a Joomla 3 fork. Most of those sites will be rebuilt on something else, or switched off when the hosting stops supporting their PHP version. UTD sites would also count as Joomla 3 in those figures, because they still report themselves as Joomla! 3.x. And W3Techs can already see them. Its [Joomla 3 version breakdown](#) on 12 October 2026 lists versions 3.11, 3.13, 3.14, 3.15 and 3.16, none of which the Joomla project ever released. Together they are a rounding error: 3.15 is 0.1% of Joomla 3 sites and the other four are each under 0.1%, against 59.0% still on 3.10.

For a Joomla 3 site that cannot be migrated yet, there is a holding position that keeps it on real Joomla, and [the other options](#) are laid out on our Joomla 3 hub. I should declare an interest, because this one is ours. The [Joomla 3.10.999 project](#) is a reference set of community security patches for stock Joomla 3.10.12, and [the mySites.guru patch tool](#) applies them with one toggle. The site still reports 3.10.12, there is no separate update server, switching it off puts the stock files back, and we describe it as a stopgap rather than a destination. [TLWebdesign](#) packages the same patches as an installable extension. None of these will save a site from its extensions, so the extension inventory still matters more than the core.

UTD's PHP 8.5 work is the part I would give it real credit for. Hosts are removing PHP 7, and on some, such as Fasthosts, switching a site to PHP 8 to test it cannot be undone, so a Joomla 3 site that is not PHP 8 compatible simply stops. Solving that solves the [hosting deadline](#), which is real and urgent. It does not solve the security one, and the two should not be confused.

I can see one group UTD is really for: sites built on K2 that never migrated to Joomla 4 because K2 never got there. That group is real, and some of it is large. The most popular sites W3Techs lists on Joomla 3 include two Greek news sites, [dnews.gr](#) and [cnn.gr](#), and the dnews.gr homepage still loads its articles through K2, with images served from K2's `/media/k2/items/cache/` folder and K2's `k2ItemsBlock` module on the page. Busy news sites like these, with years of content in K2 and no K2 for Joomla 4, are exactly the sites UTD is built around.

JoomlaWorks publishes its own numbers for that audience, and they are the most telling figures in this post. [K2 Metrics](#) is the anonymous telemetry that K2 2.7.0 and

later sends home. On 12 October 2026 it showed 4,775 K2 sites active in the last 30 days, and 4,585 of them, 96.0%, on Joomla 3. Just 33 reported a UTD version number (3.11, 3.14, 3.15, 3.16 or 3.17). That is 0.69% of K2 sites, six months after launch, among the very sites UTD exists to serve. The same telemetry shows 92.7% of those K2 sites on a PHP version older than 8.2, which means no security support from the PHP project, and 12.4% still on PHP 5. For them it buys time on a familiar stack. That is a fair reason to look at it, as long as you go in knowing you are moving onto one developer's fork, with no promise your other extensions will keep working.

If security is why you are looking at UTD, that is the strongest reason not to use it.

A Joomla 3 site on UTD gets security fixes from one developer, reviewed by no one, delivered as an unsigned zip that changes with every commit, with no private way to report a problem and no official upgrade route out. That is a less secure place for the site than the official Joomla it would have to migrate to anyway.

How mySites.guru helps with Joomla 3 sites

mySites.guru looks after portfolios of Joomla sites. Connect your Joomla 3 sites and you get the facts this post had to dig out by hand, for every site at once:

- Every site still on Joomla 3, found by the [Joomla 3 end of life check](#), with every published core vulnerability flagged against each site's exact version.
- Every vulnerable extension, from the [Joomla vulnerability scanner](#) and the [vulnerable extension list](#), which is where Joomla 3 sites are actually getting hacked.
- The community core patches in one click, with the [Joomla 3 patch tool](#) applying the 3.10.999 fixes to stock 3.10.12 and reverting them if you need to. The site stays on real Joomla, with a version number every tool understands.
- The PHP version on every site, so you know which [hosting deadline](#) you are up against before your host decides for you.

None of that replaces migrating. It tells you which sites to move first and keeps them safer while you do, and when you are ready, [moving to modern Joomla with mySites.guru](#) explains how the connector side works. [mySites.guru](#) is GBP 19.99 a month for unlimited sites, with the Joomla 3 patch tool [included in every plan](#), or you can start with a [free audit](#) of one site.

Timeline

-
- 2023-08-17** ○ **Joomla 3 reaches end of life**
Free security patches for the Joomla 3 core stop, at any severity.
 - 2024-05-30** ○ **JoomlaWorks rules out K2 for Joomla 4 and 5**
It calls Joomla 4 and 5 "failed CMS releases" and says it will work on a possible fork, starting with a rebrand.
 - 2025-02-17** ● **The paid eLTS programme ends**
The last official route to a patched Joomla 3 core closes.
 - 2026-04-20** ● **Joomla 3.x UTD 3.11 is published**
A hard fork of Joomla 3.10.20 eLTS, under the Joomla name, numbered as if it were the next Joomla release.
 - 2026-05-21** ● **UTD 3.12 adds an access-control hole**
A change to the {loadmoduleid} tag drops its access checks, so any module can be shown to anyone.
 - 2026-10-03** ● **UTD 3.16 adds the Little WAF**
Two filters, both for Regular Labs tags. Nothing for SP Page Builder.
 - 2026-10-10** ● **UTD 3.17 fixes its own hole**
The {loadmoduleid} fix is counted among 22 security fixes.

Further Reading

- [joomlaworks/joomla-3.x on GitHub](#) - the UTD source, commit history, changelog and update feed.
- [The Joomla 3.x UTD website](#) - the "secure, current & still yours" pitch in JoomlaWorks' own words.
- [K2 will not be made available for Joomla 4/5](#) - JoomlaWorks' May 2024 post announcing the fork.
- [Joomla trademark FAQ](#) - what Open Source Matters says about forks and the Joomla name.
- [Joomla 3 Didn't Fail. Your Retainer Did.](#) - our measured data on Joomla 3 sites, and every Joomla 3 rescue project compared.
- [The Joomla 3.10.999 Project](#) - the community core patches that keep a site on real Joomla 3.10.12.
- [JoomlaWorks on the Joomla 3 vendor tracker](#) - every dated JoomlaWorks statement on Joomla 3, with sources.

Frequently Asked Questions

What is Joomla 3.x UTD?

Joomla 3.x UTD is a hard fork of Joomla 3.10.20 eLTS, published by JoomlaWorks, the company behind K2, in April 2026. One developer writes it, with an AI assistant, and it has shipped versions 3.11 to 3.17 since. Open Source Matters and the Joomla project do not make, endorse or support it.

Is Joomla 3.x UTD an official Joomla release?

No. Joomla 3 ended at 3.10.x, and the Joomla project never released a 3.11 or later. UTD still reports itself as "Joomla!" with version numbers that continue Joomla's own 3.x line, which is why extensions, update servers and security scanners can mistake it for an official release.

Is Joomla 3.x UTD secure?

Its update packages are unsigned, have no checksums and are rebuilt on every push to the main branch. Its fixes are reviewed by no one outside the project, there is no private way to report a vulnerability, and version 3.12 introduced an access-control hole that was only fixed in 3.17. Official Joomla 6 has a security team, checksums and TUF-signed updates.

Can I upgrade a Joomla 3.x UTD site to Joomla 5 or 6?

Not through the one-click route. UTD points every Joomla Update channel at its own feed, and Joomla's official Joomla 4 upgrade feed only targets sites on 3.10. A site moved onto UTD's SQLite driver has to convert back with UTD's own tool first, and a SQL Server site has no conversion tool at all.

Will my Joomla 3 extensions keep working on Joomla 3.x UTD?

There is no promise that they will. UTD's minor releases have already removed core templates and drivers, its compatibility promise covers only its own past releases, and the leaner successor it is building removes com_content, which most Joomla 3 extensions depend on. We could not find a single third-party developer releasing extensions for it.

Should I use Joomla 3.x UTD or Joomla 6?

For almost every site, Joomla 6. A Joomla 3 site that cannot migrate yet can be patched in place with the community 3.10.999 core patches while it stays on real Joomla 3.10.12, and

that leaves the official upgrade route open. UTD mainly suits K2 sites that never moved past Joomla 3.

AI MODIFIED

Written and edited by a human, with AI assistance. [Our approach to AI](#)

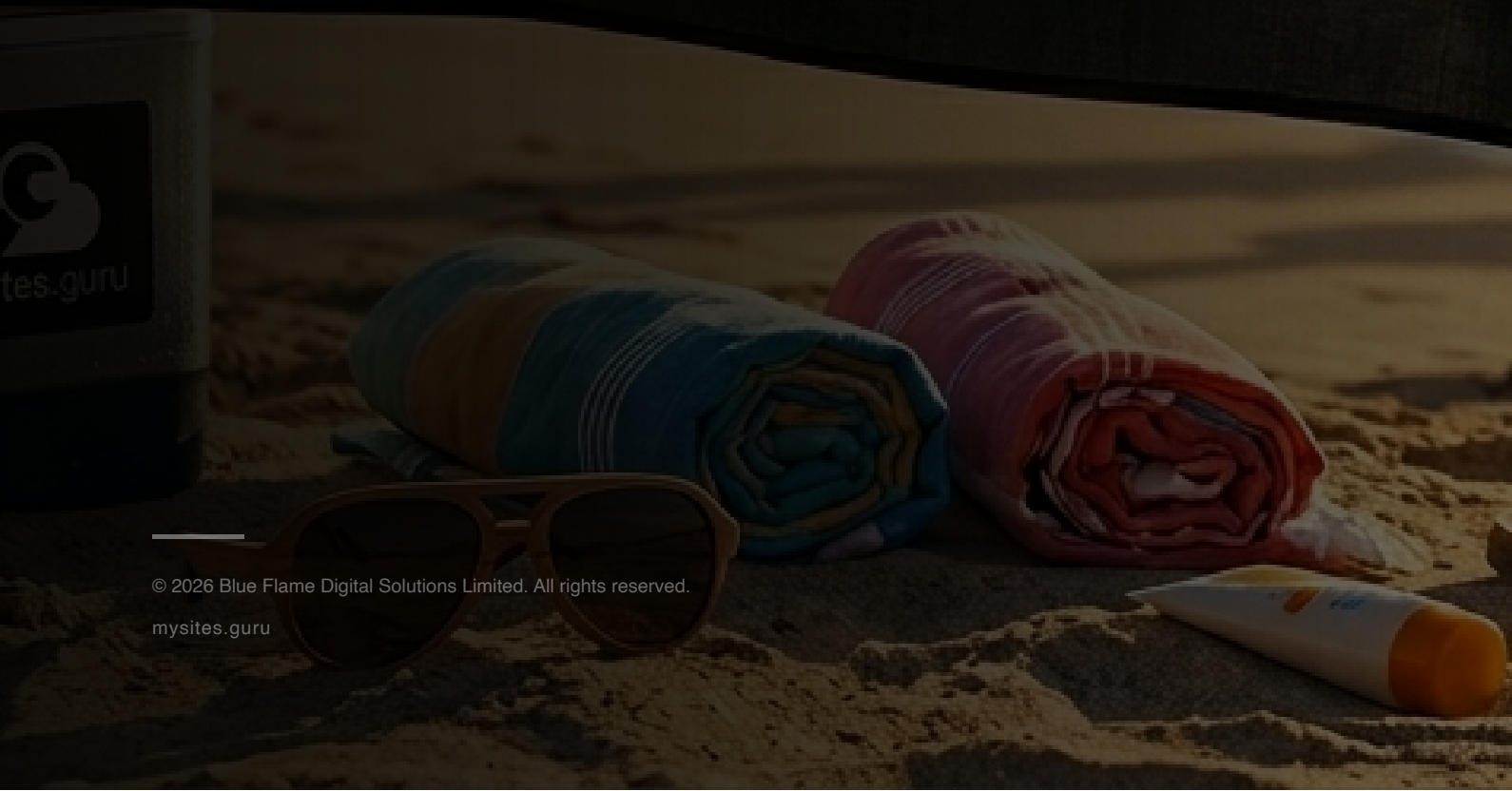
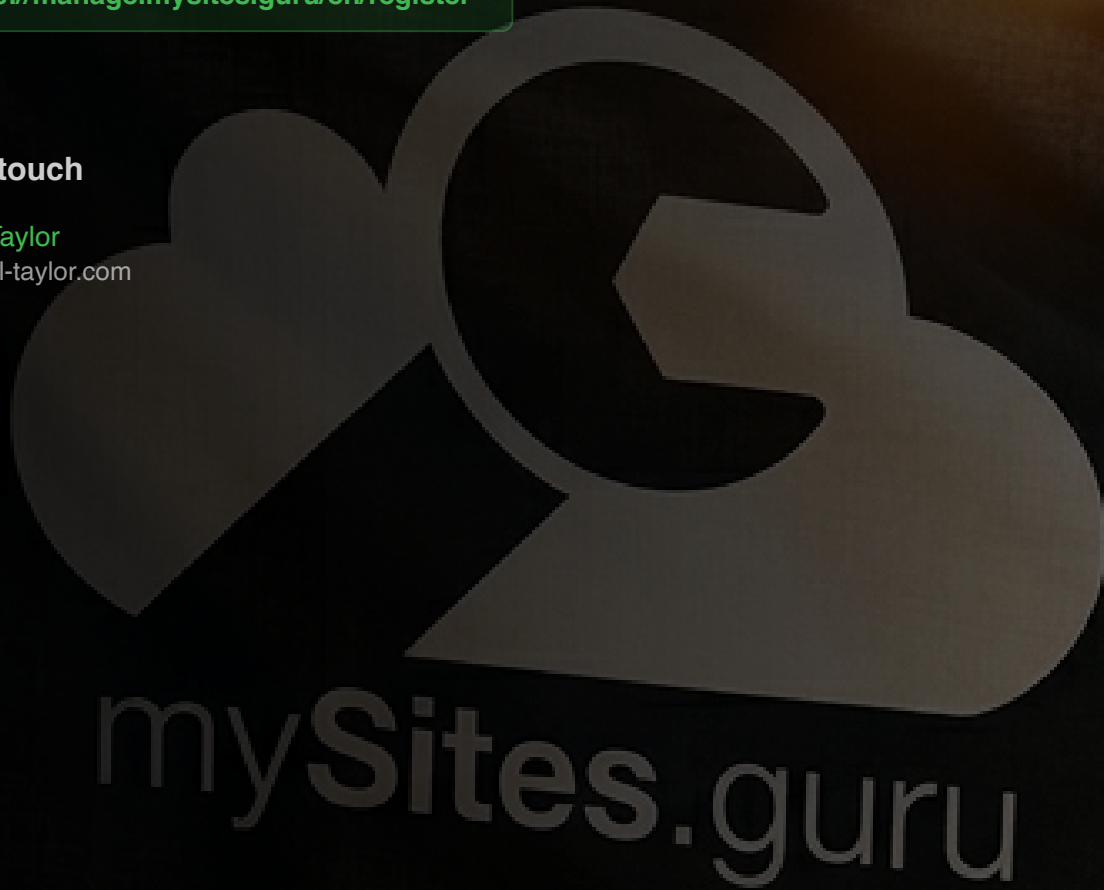
Get Your Free Site Audit

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.

mysites.guru