mySites.guru

Let's Encrypt Is Down. Renewals Are Next

Let's Encrypt halted all certificate issuance at 18:37 UTC on 2026-05-08 after a cross-signed cert problem with their new Generation Y root. Issuance resumed roughly two and a half hours later via the older Generation X root for the tlsserver and shortlived profiles.

Phil E. Taylor | 8 May 2026

● ACTIVE INCIDENT, LIVE BLOGGING

Let's Encrypt issuance is still halted as of last refresh. Refresh this page for the latest details.

We're tracking the official status page and updating this post as new information arrives. Last updated when the page was published. The incident started at 18:37 UTC on 2026-05-08.

Let's Encrypt's status page currently shows an active incident, "Stopping Issuance for Potential Incident", opened at 18:37 UTC on 2026-05-08 with a single sentence: *"We have been made aware of a potential incident and are shutting down all issuance."* The production ACME directory endpoint at `acme-v02.api.letsencrypt.org` has been returning HTTP 503 ever since. Staging is healthy. The status feed has not been updated for over an hour, there's no public root cause yet, and no ETA.

We're seeing it across mySites.guru subscriber sites already. Renewal jobs that ran in the last hour are coming back with `urn:ietf:params:acme:error:serverInternal` and `503 Service Unavailable` from `acme-v02.api.letsencrypt.org/directory`. None of those failures will reach the site owner until their alert threshold trips, or worse, until the cert actually expires and a browser starts blocking visitors. That delay is the entire reason expiry monitoring exists, and today's exactly the day it pays for itself.

The rest of this is what we know, what we don't, and what to do if you're sitting on a cert that needs to renew today.

What Does the Let's Encrypt Incident Mean for My Sites?

Existing certificates keep working until their stated expiry.

The outage breaks new issuance and renewal, not currently-valid certificates.

The TLS handshake between a visitor's browser and your web server doesn't phone home to Let's Encrypt on every request. CRLs are static-published documents. OCSP (the live revocation lookup that used to depend on CA infrastructure being available) reached end of life in August 2025. So a Let's Encrypt outage today does not in itself break sites that already hold a valid cert.

What it does break is new issuance and renewal. Every certbot, acme.sh, Caddy, Traefik, cPanel autossl and managed-host renewal job currently in flight is hitting that 503. ACME clients on default schedules typically retry with some backoff, and most are configured to renew at 60 days into a 90-day lifetime. A multi-hour outage rarely causes immediate expiry. A multi-day outage starts catching the long tail of certs that were already within their 30-day renewal window when it started.

The certs most exposed are the ones using Let's Encrypt's six-day "shortlived" profile, generally available since 2026-01-15. Shortlived certs renew every 2.5 days or so, so even a 12-hour outage eats a meaningful chunk of the renewal margin.

What's Actually Happening at Let's Encrypt?

We don't know.

Let's Encrypt hasn't published a root cause. Anything beyond "they halted issuance for a potential incident" is inference, not fact.

That's the honest answer. What we *can* say is what's technically happening, and what the public signals suggest.

Technically: the production ACME directory endpoint at `acme-v02.api.letsencrypt.org/directory` is returning HTTP 503 with `urn:ietf:params:acme:error:serverInternal`. Every certbot, acme.sh, Caddy and Traefik run is hitting that same response. Staging is healthy and returns HTTP 200, which means LE has actively cut traffic to production rather than the whole stack being broken. Cutting only production while staging keeps running is a deliberate operator action, not a crash.

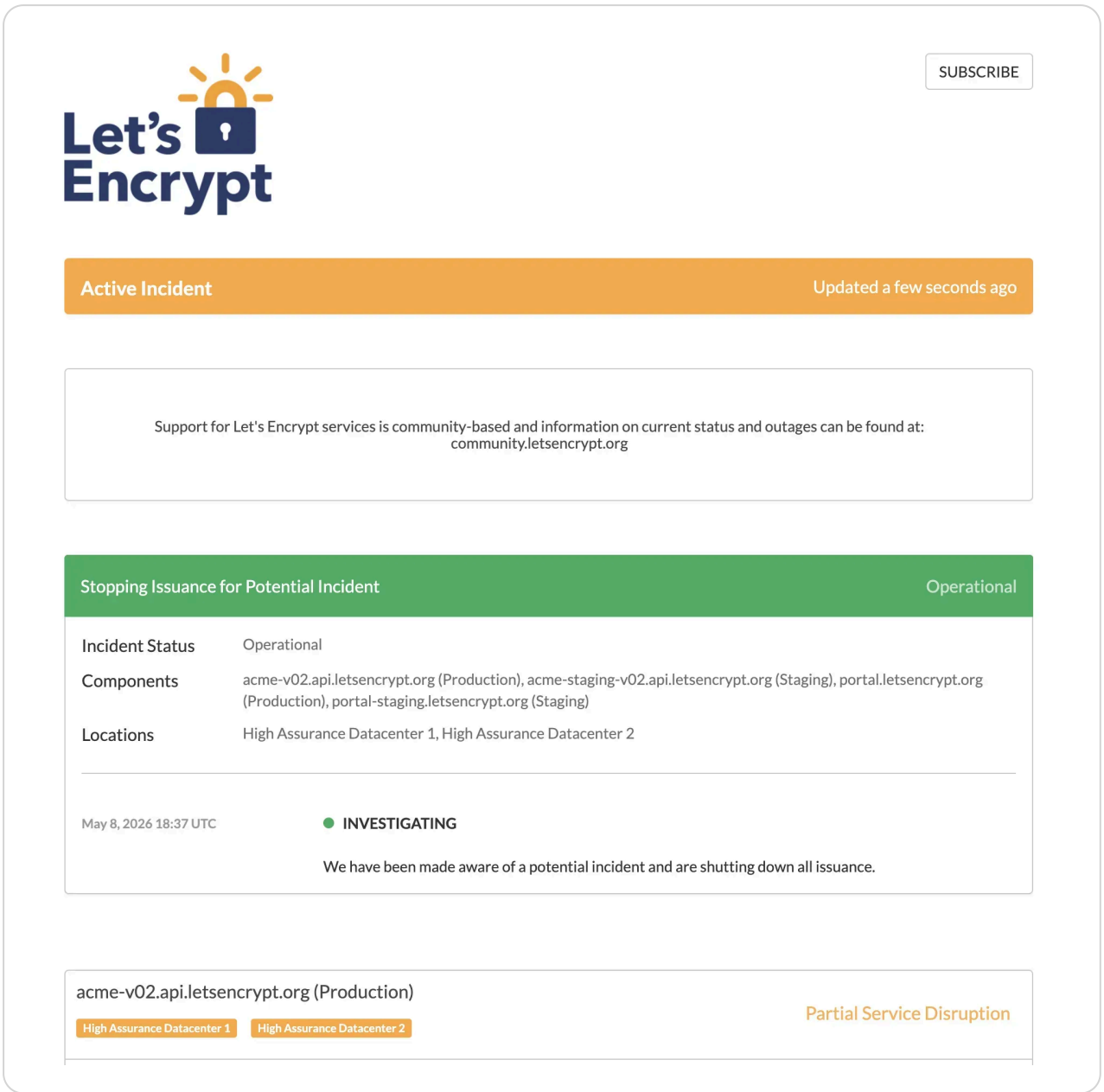
Inference, not fact: the phrase “potential incident” plus “shutting down all issuance” matches the playbook Let’s Encrypt has used before when a security or correctness concern surfaces and the safest move is to halt before you understand the cause. The 2020 CAA rechecking bug followed the same pattern: halt first, investigate, then decide whether anything already issued needs to be revoked. Our best guess is that LE has spotted a problem internally (or had one reported to them), can’t yet quantify the blast radius, and would rather stop issuing for a few hours than issue something they later have to revoke en masse.

Things we genuinely don’t know yet: whether the trigger was a software bug, an infrastructure failure, a security event, or a third-party report. Whether already-issued certificates are affected. Whether any will need to be revoked. How long the halt lasts. The community.letsencrypt.org incidents board doesn’t have a thread for it. Hacker News hasn’t picked it up at the time of writing. Twitter chatter is starting but unconfirmed.

Until LE publishes more, this is a wait-and-see for most sites and an act-now for any site that has to renew in the next few hours.

Incident Timeline

All times UTC. We’re updating this list as new signals come in.



Time (UTC)	What happened
2026-05-08 18:37	Let's Encrypt opens incident **"Stopping Issuance for Potential Incident"**. Single-sentence message: *"We have been made aware of a potential incident and are shutting down all issuance."* Components flagged: acme-v02.api.letsencrypt.org (Production), acme-staging-v02 (Staging), portal.letsencrypt.org, portal-staging.
2026-05-08 18:47	Status page last updated. Production confirmed at **"Partial Service Disruption"**, datacenters High Assurance 1 and 2. No further public communication after this point.

Time (UTC)	What happened
~19:00	First failed renewal jobs on mySites.guru subscriber sites. ACME clients return `urn:ietf:params:acme:error:serverInternal` against `/directory`. Staging endpoint still healthy, returning HTTP 200, confirming the halt is targeted at production.
~19:30	Production `/directory` still HTTP 503. No thread on community.letsencrypt.org incidents board. No top-of-page Hacker News discussion. Twitter chatter starting but unconfirmed.
Now	Watching letsencrypt.status.io and community.letsencrypt.org/c/incidents for any update. Will append rows as new information lands.

> **Warning**
> If you have already-issued certs that need re-issuance for any reason today (key compromise, domain change, SAN addition), assume that operation is blocked until further notice and plan around it.

How Do I Find Every Site With an At-Risk SSL Certificate?

The hard part of an incident like this isn't the technical fix on any one site. It's knowing which sites in your portfolio are at risk. If you manage twenty Joomla and WordPress sites for clients, you have two questions to answer in the next ten minutes:

1. Which of my sites' Let's Encrypt certs are inside their renewal window today?
2. Which of my sites' renewal crons ran in the last hour and silently failed?

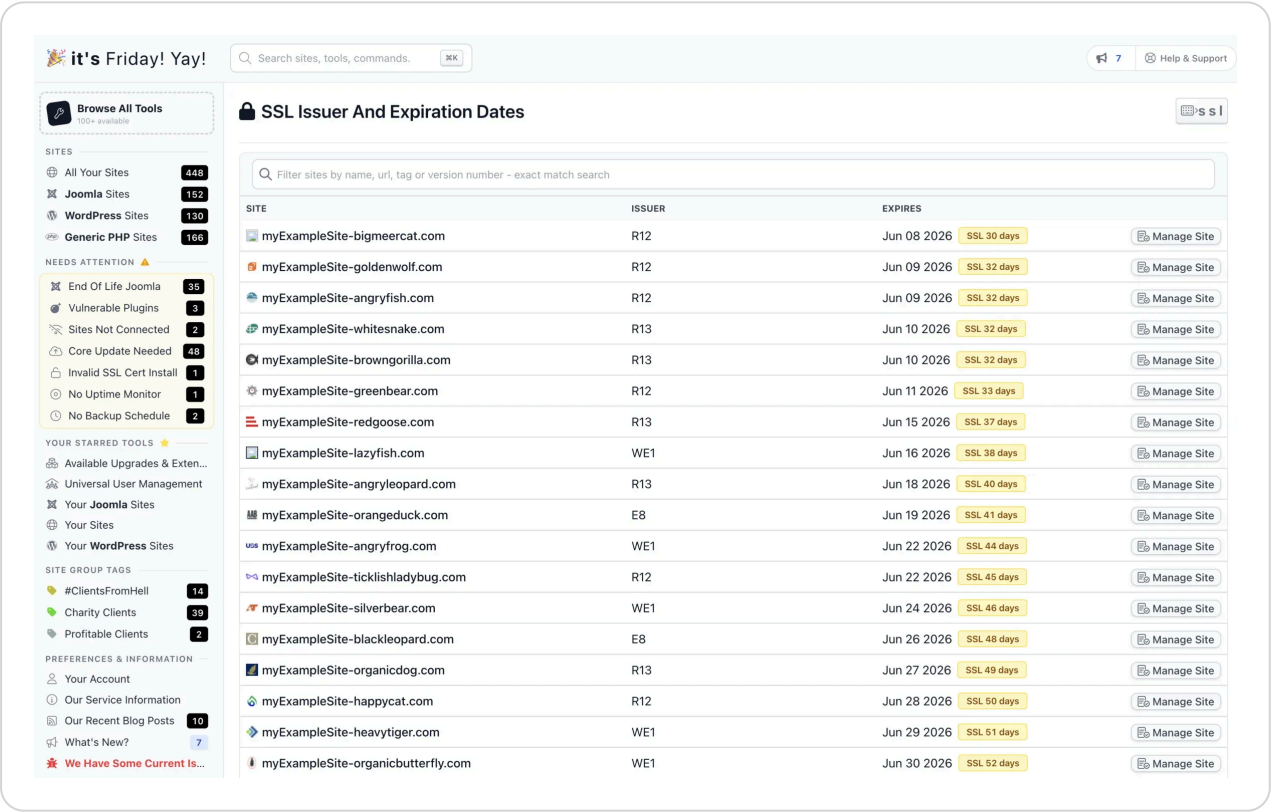
mySites.guru has tracked SSL certificates on every site snapshot since 2012. On every snapshot we download the active certificate the same way a browser would, then record the issuer, expiry date and full chain validity. The dashboard sorts every connected site by certificate expiration date. During an incident like today's, that sorted list is your priority queue.

The default alert window is two days before expiry. For most sites running a healthy renewal cron that's plenty (the cron renews at day 30, the alert is a safety net). Today is the day that safety net actually matters.

If you're not a subscriber yet, you can connect your sites and run the same SSL audit in a free audit. The SSL state of every connected site is one of the things the audit covers.

What Can mySites.guru Subscribers Do Right Now?

Open manage.mysites.guru/en/sites/by/ssl/expiration. The page lists every connected site sorted by SSL expiration date, with the issuer (Let's Encrypt, ZeroSSL, Sectigo, cPanel auto-SSL, whatever) for each. That's the single view you want during this incident.



Reading the issuer column

The codes in the Issuer column are the certificate authority's intermediate CA short names. The ones starting with **R** (R10, R11, R12, R13, R14) and **E** (E5, E6, E7, E8, E9) are all Let's Encrypt: R-series are RSA intermediates, E-series are ECDSA. **WE1** and **WR1-WR4** are

Google Trust Services. Anything starting with R or E in this dashboard means the site is renewing through Let's Encrypt and is exposed to today's incident.

Three things to do with it:

1. Anything in the next 14 days needs eyes on it today. Sort by expiration, look at the top of the list, decide whether each site can wait or needs a manual intervention (paid stopgap cert or a switch to ZeroSSL/Buypass).
2. Filter by issuer. If a site is showing a non Let's Encrypt issuer, it's not affected by this incident at all. Move on.
3. For at-risk sites, check the renewal cron logs on the host. Most failures right now will be `urn:ietf:params:acme:error:serverInternal` from `acme-v02.api.letsencrypt.org/directory`. That's the LE outage, not anything wrong with your config.

If your alert window is set at the default two days, push it out for the duration of the incident. We'd rather over-alert and have you ignore a few than under-alert and miss the one site that mattered. The alert window is configurable per site.

Export the list to CSV from the same screen if you want to share it with a co-worker or work the priority queue offline.

How Do I Renew an SSL Certificate When Let's Encrypt Is Down?

If you have a cert that genuinely needs to renew before Let's Encrypt is back, there are three options that actually work.

The first is to switch your ACME client to a backup CA. ZeroSSL and Buypass both speak ACME, both issue free DV certificates, and both are commonly used as Let's Encrypt alternatives.

- ZeroSSL issues 90-day DV certs via ACME, supports wildcards, and requires External Account Binding (EAB). EAB is a one-time pairing step: log into ZeroSSL

Developer, click "Generate" under EAB Credentials for ACME, and copy the two strings it gives you (a KID and an HMAC key). The KID is short. The HMAC key is a longer base64 string. You only generate them once per account and reuse them for every certbot run.

- Buypass Go SSL issues 180-day DV certs via ACME, no EAB required, no wildcard support.

For certbot, the switch looks roughly like the example below. **This is illustrative, not copy-and-paste.** Replace the placeholders with your real EAB credentials, your webroot path, and your domain names before running it.

```
# EXAMPLE - swap placeholders for real values before running
certbot certonly \
  --server https://acme.zerossl.com/v2/DV90 \
  --eab-kid PASTE_YOUR_EAB_KID_FROM_ZEROSSL_DASHBOARD \
  --eab-hmac-key PASTE_YOUR_EAB_HMAC_FROM_ZEROSSL_DASHBOARD \
  --webroot -w /var/www/html \
  -d example.com -d www.example.com
```

`--eab-kid` and `--eab-hmac-key` come from the ZeroSSL EAB credentials screen. `-w` is the webroot path that serves files for `example.com` (whatever directory `/.well-known/acme-challenge/` is reachable from on your host). `-d` lists every name you want on the cert. If you're on Buypass, the command is the same but with `--server https://api.buypass.com/acme/directory` and no EAB flags.

Two caveats. If you have CAA DNS records pinning issuance to `letsencrypt.org`, the new CA's challenge will fail until you add or replace the CAA record. And every CA has its own rate limits, so don't try to migrate hundreds of sites in one batch. A handful of high-priority renewals is the right scope during a short outage.

The second option is a paid cert as a stopgap. DigiCert, Sectigo, GlobalSign, Namecheap and Google Trust Services all issue paid 1-year DV certs for somewhere between $5 and $50 with same-day turnaround. Not elegant, but if a flagship site is

hours from expiry and you need certainty today, this is the option that delivers it. Replace it with Let's Encrypt once issuance resumes.

The third option is to wait. For most sites that's the right answer. Let's Encrypt outages have historically resolved in hours, not days. Default 90-day certs renewing at day 30 have a 30-day cushion. Unless you're on shortlived certs or you're already inside the 14-day red zone, waiting is fine.

> **Note**
>
> If your renewal cron is going to retry every hour for the next week regardless of whether it succeeds, you do not need to do anything special during the outage. The retry will eventually succeed. The risk is the silent-fail case where the cron logs an error and nobody reads the log.

Why Do Short Certificate Lifetimes Make Outages Worse?

Let's Encrypt's original 90-day rationale from 2015 was straightforward: shorter lifetimes limit the damage of a key compromise and force operators to automate renewal. Both arguments are correct. The trade-off, less discussed at the time, is that the operational margin for any issuance disruption is exactly the unused portion of the certificate lifetime at the moment the disruption starts.

In December 2025 Let's Encrypt announced its intention to drop the default lifetime from 90 days to 45 days by 2028. In January 2026 the six-day shortlived profile reached general availability. Both moves are defensible on security grounds (shorter lifetimes do reduce compromise windows). The cost they impose is sensitivity to outages like today's. A 90-day cert renewing at day 30 has a 30-day buffer. A 45-day cert renewing at day 15 has 15 days. A 6-day shortlived cert renewing every 2.5 days has hours.

Short-lived certs aren't bad. They're just unforgiving. As the industry moves towards them, the cost of any single CA going down for half a day grows. Roughly 63% of all TLS-enabled sites depend on Let's Encrypt, which makes the short-lifetime trend and

the CA monoculture problem interact badly. Today's incident is a low-stakes preview of what a longer outage will feel like once 6-day certs are common.

How Has Let's Encrypt Recovered From Past Incidents?

A short operator memory of recent Let's Encrypt history:

- 2020-02-29: the CAA rechecking bug. A Boulder bug let CAA records be bypassed in some renewal flows. 3,048,289 certificates were flagged. Let's Encrypt eventually decided not to revoke ~1 million of them past the deadline because doing so would have broken too much of the web. Operational scramble lasted about five days. Bugzilla #1619179.
- 2025-07-21: complete API outage. Multi-hour total ACME API failure, postmortem dated 2025-09-02. No mass revocation, but widespread renewal failures.
- 2025-08-06: OCSP responders shut down for good. Not an outage, but it changed the failure mode for every Let's Encrypt incident going forward. There's no live revocation service for browsers to fail to reach. CRLs are now the only revocation transport, and CRLs are static.
- 2026-01-15: six-day shortlived certs GA. Increased the population of certs that are exposed to short outages.

Today's incident is in the "unknown duration, unknown blast radius" bucket until Let's Encrypt says more. Based on the historical pattern, a few-hours resolution is plausible. A multi-day resolution would be exceptional but not unprecedented.

What Should I Do Right Now?

If you manage one or two sites: log in, check whether your cert expires in the next 14 days, and if so set a calendar reminder to verify renewal once Let's Encrypt is back.

If you manage many sites: pull a single dashboard view of every site's SSL expiry. Sort by date. Anything within 14 days is your priority queue. mySites.guru's SSL monitoring

view does this; so does any equivalent tool. The tool isn't really the point. Having the answer in seconds, rather than clicking through wp-admin tabs while the clock runs, is.

If you have a cert that absolutely cannot wait: switch ACME clients to ZeroSSL or Buypass, update CAA records if needed, accept the EAB key configuration overhead. Or buy a paid 1-year cert as a stopgap. Both are fine. Don't try to migrate your whole portfolio at once.

Whatever happens with today's incident, the underlying lesson is the same one Let's Encrypt's roadmap has been pushing operators towards for a decade. Certificate monitoring stops being optional the day a CA monoculture meets short-lived certs. The combination is too efficient on the happy path and too brittle on the failure path to operate without alerting in place.

Further Reading

- Let's Encrypt status page, live incident updates
- Let's Encrypt incidents category, official postmortems
- Why 90-day lifetimes? (Let's Encrypt, 2015)
- From 90 to 45: cutting default cert lifetime (Let's Encrypt, 2025)
- Six-day and IP-address certificate GA (Let's Encrypt, 2026)
- ZeroSSL ACME documentation
- Bugzilla #1619179: 2020 CAA rechecking bug

We'll update this post once Let's Encrypt publishes a root cause and an all-clear. If you want a single view of SSL expiry across every Joomla and WordPress site you manage, the free audit is the fastest way to get one in front of you.

Frequently Asked Questions

What is the Let's Encrypt incident on 2026-05-08?

Let's Encrypt halted all certificate issuance at 18:37 UTC on 2026-05-08, posting only that they had been 'made aware of a potential incident'. The acme-v02 production directory endpoint has been returning HTTP 503 ever since. Staging is unaffected. As of writing the status feed has not been updated for over an hour and no root cause has been disclosed.

Will my existing SSL certificate stop working during a Let's Encrypt outage?

No. Existing certificates keep working until their stated expiry date. Browsers do not phone home to the issuing CA on every TLS handshake, and Let's Encrypt shut down OCSP responders in August 2025, so there is no live revocation lookup that could fail. The risk is renewal: any cert within ~30 days of expiry that needs to renew during the outage will fail to renew.

How long does Let's Encrypt take to recover from outages?

Historically most ACME outages resolve within a few hours. The 2020 CAA rechecking bug was a multi-day operational scramble that affected 3 million certificates. The 2025-07-21 complete API outage required a multi-day post-mortem. Today's incident is in the 'unknown duration' category until they say more.

What can I do if a Let's Encrypt renewal fails right now?

Three options. Wait it out if the cert is more than a few days from expiry. Switch the ACME client to ZeroSSL or Buypass for the affected site, both of which are free and ACME-compatible. Or, if the site is high-stakes and expiry is hours away, buy a paid 1-year cert from a commercial CA as a stopgap. Update CAA DNS records first if they limit issuance to letsencrypt.org.

What share of the web depends on Let's Encrypt?

Let's Encrypt issues approximately 10 million certificates per day and accounts for around 62.7% of all websites with TLS, per W3Techs (May 2026). It is the dominant CA for free DV certificates. Cloudways, cPanel autossl, Plesk, Caddy and most managed-hosting platforms default to Let's Encrypt, so an issuance halt cascades through nearly every category of site.

Are 6-day Let's Encrypt certificates affected differently?

Yes, much more severely. Let's Encrypt's six-day shortlived certificate profile went generally available on 2026-01-15. Sites using shortlived certs renew every 2.5 days or so. A multi-

hour outage eats most of the renewal buffer. Anyone who opted into shortlived certs has very little margin during today's incident.

Does mySites.guru alert on SSL certificate failures during incidents like this?

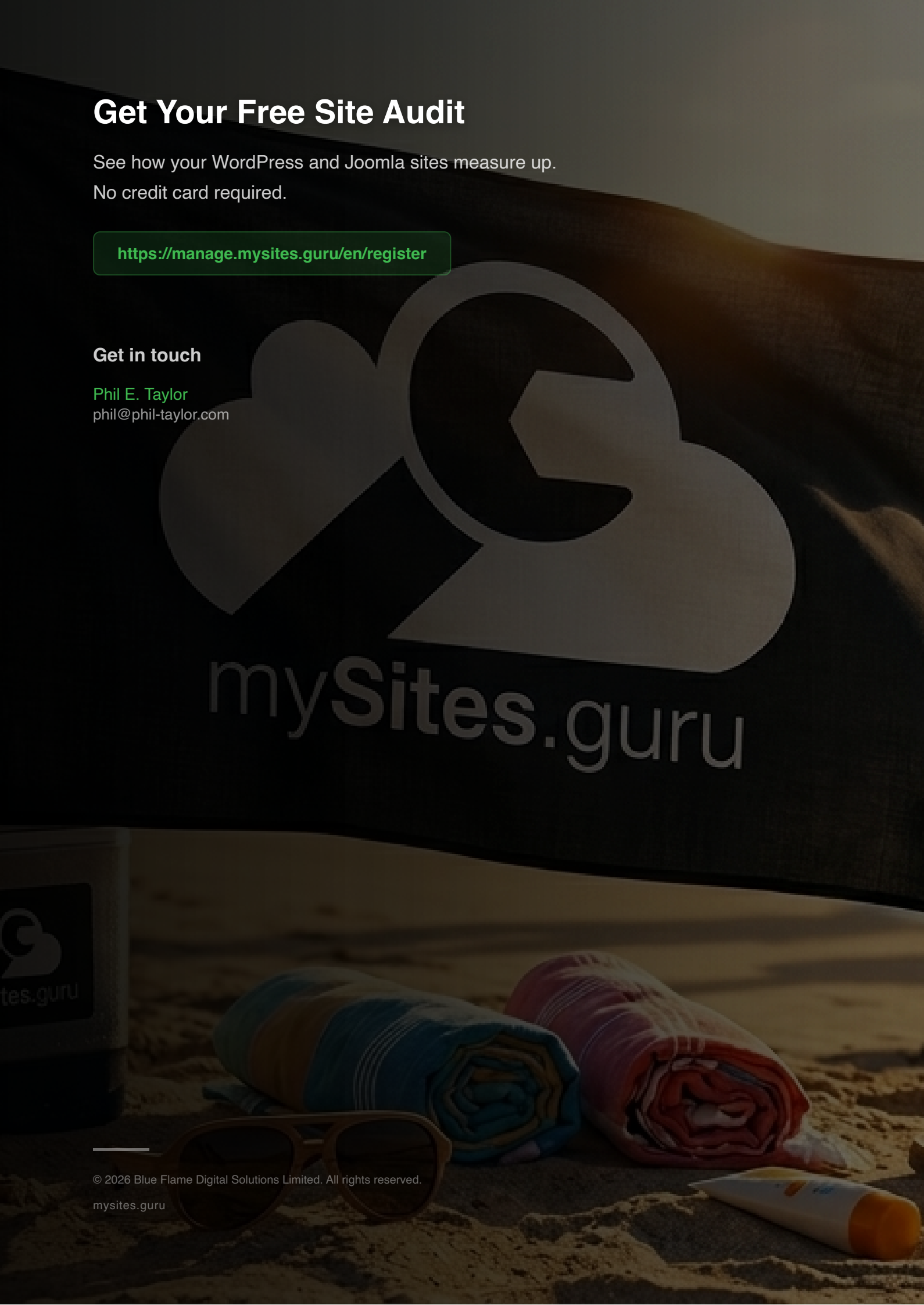
Yes. mySites.guru has tracked SSL issuer, expiry date and full chain validity on every site snapshot since 2012. Default alert is two days before expiry, configurable. During incidents like today's, the alert acts as the early-warning that a renewal cron has silently started failing.

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