VULNOW Q3 2026 VULNERABILITY DISCLOSURE REPORT

Before the Advisory, Before the Exploit

VulNow flagged 1,294 open source vulnerabilities before they were publicly disclosed in the third quarter of 2026. Four were later exploited in the wild, and VulNow flagged each one before its advisory was published and before it was listed as exploited.

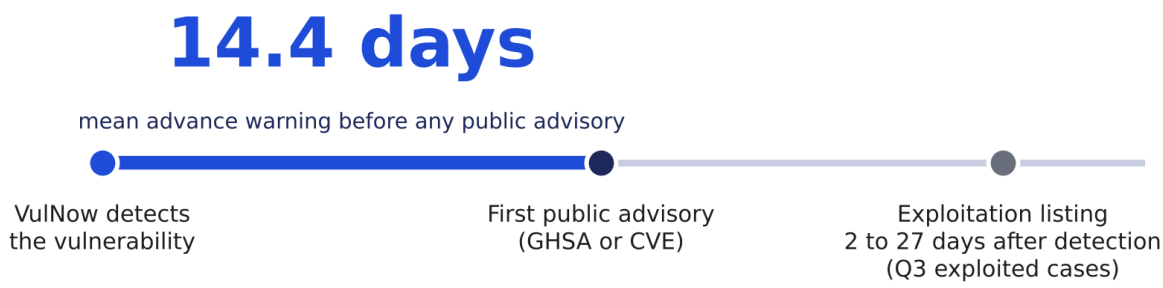


Figure 1. Where VulNow's signal sits relative to the public advisory and the exploitation listing. Not to scale.

> *"The advisory is the last to know."*
>
> **Cassie Crossley, CEO and Co-Founder, VulNow**

What is a PreCVE?

A PreCVE is a vulnerability in open source code that VulNow identifies before any public security advisory exists for it. Most begin as a silent fix: a maintainer patches the flaw in a public code change without announcing it as a vulnerability.

A PreCVE is **confirmed** when a GitHub Security Advisory (GHSA) or CVE record is later published for it. The time between VulNow's detection and that first public advisory is the **lead time**: the advance warning a security program would have had.

Contents

Executive Summary 3
Key Findings 4
The Window Before the Advisory 4
Exploited Before Anyone Was Told 7
EU CRA Reporting Is Now Live 9
Can You Trust the Signal? 10
What We Cover 12
What This Means for a Security Program 14
External sources 14

About VulNow

VulNow B.V. is a predictive software supply chain risk intelligence platform based in Amsterdam. VulNow identifies undisclosed vulnerabilities, silent patches, and Dark Matter Vulnerabilities™ in open source code before public advisory publication, and delivers that intelligence to enterprise defenders and to security vendors who embed it in their own products. VulNow is a CVE Numbering Authority under the ENISA CVE Root.

The figures in this report reflect the platform as of September 30, 2026. In late October 2026, more than 3,000 VulNow findings have been confirmed by a published security advisory. The confirmed PreCVE dataset is public, anonymous, and read only at https://precve.vulnow.com, with an OSV format API for programmatic access.

For more information, contact info@vul.now, or Cassie Crossley, CEO / Co-Founder, or Valerio Mulas, CTO / Co-Founder, directly.

Executive Summary

By September 30, 2026, **1,864 VulNow findings** had been confirmed by a public security advisory published after our detection. On average, VulNow flagged the findings **14.4 days** before any public advisory existed.

1,864 PreCVEs confirmed by a published advisory (cumulative)	**14.4 days** Mean lead time before the first public advisory	**4.8%** False-positive rate across analyzed findings	**9** Flagged early and later exploited in the wild (cumulative)
~20,000 Open source repositories monitored	**870** Distinct open source packages with confirmed findings	**12** Ecosystems, including npm, Maven, PyPI, and Go	**186 days** Longest lead time at quarter end (react-router)

Q3 2026 at a glance. VulNow platform data as of September 30, 2026.

- **Anticipation.** 1,294 vulnerabilities disclosed in Q3 had already been flagged by VulNow before any public advisory existed. By the advisory's own rating, 93 were Critical and 508 High.
- **Exploitation.** Four Q3 findings were later confirmed to be exploited in the wild (Rails, Dolibarr, GitLab, WordPress). We flagged each one before disclosure and before it was listed as exploited.
- **Precision.** 4,674 findings had been analyzed by September 30th. 4,406 were real vulnerabilities and 225 were false positives (4.8%), typically code fixes that turned out not to be security fixes.
- **Coverage.** Confirmed findings span 870 distinct open source packages across 12 ecosystems, from npm, Maven, and PyPI to Go modules and Linux distributions, up from 97 packages across six ecosystems in Q2.

Key Findings

The eight findings below are drawn directly from the Q3 data and may be quoted with attribution to the VulNow Q3 2026 Vulnerability Disclosure Report.

1	**Four vulnerabilities later exploited in the wild, in WordPress, GitLab, Rails, and Dolibarr, were flagged by VulNow before their advisories and before their exploitation listings.** Rails was flagged 27 days before it was listed as exploited.
2	**The exploitation listing that most programs use to escalate arrived 2 to 27 days after VulNow's signal.** Known Exploited Vulnerabilities (KEV) catalogs record exploitation that has already been observed. They are confirmation, not warning.
3	**More than half of confirmed vulnerabilities were known more than a week before any public advisory.** 53 percent of 1,864 confirmed PreCVEs were flagged over a week ahead; 81 percent more than a day ahead.
4	**On average, a confirmed vulnerability was detectable 14.4 days before the advisory. Under the EU CRA, manufacturers have 24 hours to issue an early warning once exploitation is known.** CRA reporting obligations became applicable on September 11, 2026.
5	**One react-router vulnerability was flagged 190 days before the CVE or 186 days before the GHSA was published.** CVE-2026-53668, advisory published July 23, 2026.
6	**VulNow flagged 601 High or Critical vulnerabilities before they were publicly disclosed in Q3.** Severity as rated by the published advisory: 93 Critical and 508 High, among 1,294 Q3 disclosures VulNow had already flagged.
7	**GitHub and the NVD agree on a vulnerability's severity only about half the time.** Measured on VulNow's findings with a published CVE.
8	**Fewer than 5 in 100 analyzed VulNow findings were false positives.** 225 of 4,674 analyzed findings (4.8 percent), typically ordinary code fixes rather than security fixes.

The Window Before the Advisory

Lead time is measured from VulNow's detection to the first public advisory of any kind, whether a GitHub Security Advisory (GHSA) or a CVE record. During that window the vulnerability is real and usually already fixed upstream, but it is invisible to every tool that matches against published advisories.

By September 30th, 1,864 VulNow findings had been confirmed by an advisory published after our detection. 1,294 of them were disclosed during Q3.

How much warning

81 percent of confirmed findings arrived more than a full day before any public advisory, and **53 percent** more than a week ahead. The mean lead time is 14.4 days, up from about 10 days in Q2. The longest window at quarter end was react-router (CVE-2026-53668), flagged 190 days before the CVE or 186 days before the GHSA was published on July 23.

> *"Most security programs start the clock when a CVE is published. The real exposure starts when the fix ships. We measured that gap at 14.4 days on average and more than six months at the extreme, and attackers are already working in it."*
>
> **Valerio Mulas, CTO and Co-Founder, VulNow**

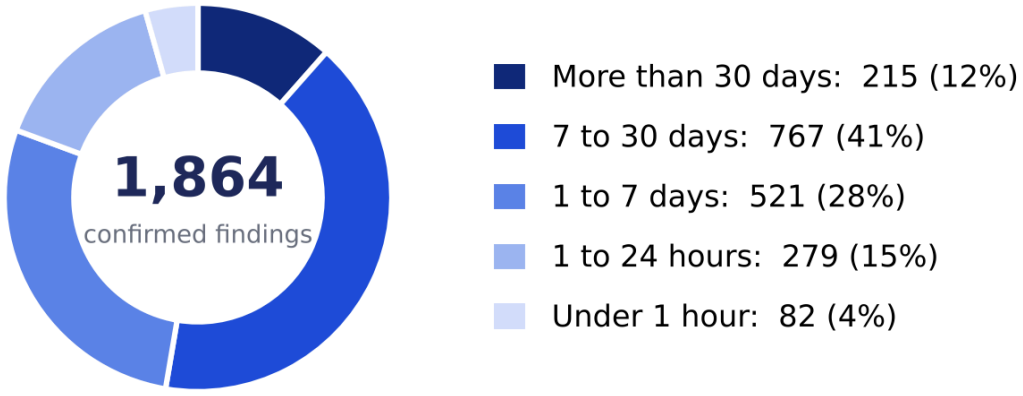


Figure 2. Confirmed findings by advance warning before the first public advisory, September 30, 2026.

How severe

These are not minor issues. Of the 1,294 vulnerabilities disclosed in Q3 that VulNow had already flagged, the advisories themselves rated 93 Critical and 508 High.

Table 1 shows the severity of all 1,864 confirmed findings in two ways: the rating VulNow assigned at detection, and the rating in the published advisory once it appeared. By the advisory's own rating, 780 confirmed findings are High or Critical. The "Not rated" row is discussed later in this report.

Severity	VulNow at detection	Share	Published advisory	Share
Critical	148	8%	133	7%
High	1,083	58%	824	44%
Medium	509	27%	747	40%
Low	123	7%	107	6%
Not rated	1	0%	53	3%

Table 1. Severity of confirmed findings at detection and in the published advisory.

Why the window is not closing

The length of the window is set largely by the public disclosure system, and that system remained under strain through Q3.

- **GitHub Advisory Database.** In June, GitHub reported that publication times had stretched from about a week to multiple weeks for a meaningful share of advisories after record submission volumes. By October, researchers in GitHub's community forum were describing a turnaround of about five weeks as the current norm, with some advisories waiting considerably longer.
- **National Vulnerability Database (NVD).** NIST began enriching only a prioritized subset of CVEs in April 2026. A U.S. Department of Commerce Inspector General evaluation found that NIST would be unable to clear its backlog of unprocessed vulnerabilities or prevent future processing delays without significant changes.

Every week added to these queues is a week added to the window in which a vulnerability is known to someone, often fixed in a public commit, and absent from the data that scanners and vulnerability management platforms consume.

Sources: GitHub, June 29, 2026; GitHub Community discussion #210051, October 2026; NIST, April 2026; U.S. Department of Commerce OIG, OIG-26-020-I, 2026. Full references in the final section.

Exploited Before Anyone Was Told

Known Exploited Vulnerabilities (KEV) catalogs list vulnerabilities for which there is evidence of exploitation in the wild. The best known is maintained by CISA, the U.S. Cybersecurity and Infrastructure Security Agency. VulnCheck maintains a broader catalog that often lists exploitation earlier. Many security programs treat a KEV listing as the trigger to escalate a vulnerability to the top of the patch queue.

In Q3, four VulNow findings were later confirmed to be exploited in the wild. Each was flagged before public disclosure, and each was flagged before it appeared in a KEV catalog.

CVE	Project	Ahead of disclosure	Ahead of KEV listing
CVE-2026-87902	WordPress	2 hours	3 days (CISA KEV)
CVE-2026-89013	Dolibarr	5 days	11 days (VulnCheck KEV); also reproduced by VulNow
CVE-2026-19478	GitLab	14 minutes	2 days (VulnCheck KEV)
CVE-2026-66066	Rails	1 day	27 days (VulnCheck KEV)

Table 2. Q3 findings later confirmed exploited in the wild.

Two features of these cases matter for prioritization. First, the lead time ahead of disclosure varies widely, from 14 minutes to five days, but the lead time ahead of the KEV listing is consistently measured in days. In every Q3 case, the listing most programs use to escalate trailed VulNow's detection by between two and 27 days. Second, these are widely deployed platforms rather than obscure libraries. WordPress, GitLab, and Rails sit at the core of many organizations' internet-facing estates.

With the five cases identified in the first half of the year, V8 (two), Apache ActiveMQ (two), and Starlette, nine VulNow findings have now been flagged before disclosure and later exploited in the wild. Starlette was flagged 103 days ahead of its CISA KEV listing.

> *“By the time a vulnerability shows up on an exploited list, it's not a warning anymore, it's a confirmation. In every exploited case this quarter, we had the signal hours or days before any reporting platform. That's the difference between patching on your schedule and patching on the attacker's.”*
>
> **Cassie Crossley, CEO and Co-Founder, VulNow**

Ransomware prediction: the Starlette case

One exploitation listing from this quarter originates from an early signal captured during Q2. On May 22nd, VulNow detected CVE-2026-48710, an authentication bypass involving Host headers in Starlette, the core ASGI library underlying FastAPI alongside numerous Python AI gateway frameworks. This alert arrived four days prior to its NVD entry on May 26 and thirteen days ahead of its GitHub advisory on June 4, with internal verification and mapping completed by June 9.

The significance of this early detection became apparent in Q3: CISA placed the CVE into its KEV catalog on September 2, **103 days after initial identification**. Furthermore, active exploitation pairing **CVE-2026-42271** with **CVE-2026-48710** was attributed to Qilin ransomware threat actors targeting public AI infrastructure for unauthenticated execution (according to Wiz and VulnCheck).

With approximately **500 million monthly downloads,** Starlette demonstrates VulNow's ability to uncover critical, ransomware-connected exposure an entire quarter before formal confirmation.

Timeline

- **May 22.** VulNow flags CVE-2026-48710 in Starlette directly from the upstream commit; initial rating of Medium.
- **May 26.** Entry recorded in NVD and listed by VulnCheck KEV on the same date, yielding a 4-day lead over public disclosure.
- **June 4.** Publication of the GitHub Security Advisory.
- **August 27.** Wiz connects active exploitation of the combined CVE-2026-42271 and CVE-2026-48710 chain to Qilin ransomware; VulnCheck adds ransomware tag based on Wiz analysis.
- **September 2**. CISA includes CVE-2026-48710 in its Known Exploited Vulnerabilities catalog, 103 days following detection.

KEV catalogs have a structural floor

KEV catalogs are among the most useful prioritization signals available, but they are curated records of what has already been observed. CISA's catalog requires an assigned CVE ID before a vulnerability can be added, along with reliable evidence of exploitation and clear remediation guidance. A vulnerability that is being exploited but has only a GitHub Security Advisory, or no advisory at all, cannot appear there yet.

Our year-end report will examine every exploited case in the 2026 dataset, including the timing between detection, disclosure, and listing.

EU CRA Reporting Is Now Live

On September 11, 2026, the reporting obligations in Article 14 of the EU Cyber Resilience Act (CRA) became applicable, and ENISA launched the Single Reporting Platform through which notifications are made. The rest of the regulation applies starting December 11, 2027.

Manufacturers of products with digital elements on the EU market must now report any actively exploited vulnerability in their products, and any severe incident affecting product security. The obligation covers products already on the market, regardless of when they were sold. For an actively exploited vulnerability, the timeline runs from the moment the manufacturer becomes aware:

- **Early warning** to the coordinating CSIRT and ENISA within 24 hours.
- **Vulnerability notification** within 72 hours, with details of the product, the nature of the exploit, and corrective measures.
- **Final report** within 14 days after a corrective or mitigating measure is available.

Non-compliance can attract administrative fines of up to EUR 15 million or 2.5 percent of total worldwide annual turnover, whichever is higher.

Why this changes the value of lead time

Most commercial products are built substantially from open source components. When one of those components is exploited, the manufacturer has to determine quickly whether its products include the affected versions, whether the vulnerable code is reachable, and what the fix is. Under a 24-hour clock, that work cannot wait for an advisory to be published, enriched, and imported into a scanner.

The Q3 exploited cases show how that plays out. In each, VulNow had identified the vulnerable component, the affected versions, and the fixing change before the public advisory existed. A manufacturer with that information has done most of its assessment before exploitation is ever reported. A manufacturer without it only begins the assessment once exploitation is reported, with the 24-hour clock already running.

> *“Since September 11, 2026, manufacturers selling into Europe have 24 hours to raise the alarm on an exploited vulnerability. Our data shows the average vulnerability was knowable two weeks before its advisory. Regulators have shortened the clock. The disclosure system hasn't caught up.”*
>
> **Cassie Crossley, CEO and Co-Founder, VulNow**

This section summarizes public guidance on Regulation (EU) 2024/2847 and is not legal advice. Sources: European Commission, CRA reporting; Regulation (EU) 2024/2847.

Can You Trust the Signal?

Early warning is only useful if it is accurate. A feed that raises more noise than signal adds to the triage burden it is meant to reduce. This section sets out what we know about precision and severity, and the limits of what we know.

Precision

Every VulNow finding either has an outcome or is waiting for one. A finding is **analyzed** once it has an outcome: a matching official advisory was published, or the finding was reviewed against the code change itself. A finding that is **not yet analyzed** has neither; most are recent findings still waiting for an advisory to appear. Most analysis is automatic, so the analyzed share grows as disclosures catch up.

As of September 30th, **4.8 percent** of analyzed findings were false positives: 225 of 4,674. A false positive is almost always a code change that looked like a security fix but turned out to be an ordinary code fix.

Status at September 30	All VulNow findings	Created in Q3
Real vulnerability (verified or confirmed)	4,406	3,207
False positive (code fixes that are not vulnerabilities)	225	128
Disputed	43	34
Analyzed (total of the rows above)	4,674	3,369
Not yet analyzed	39,250	34,595
False-positive rate	4.8%	3.8%

Table 3. Analysis status of VulNow findings on September 30, 2026.

We want to be precise about scope. About 11 percent of all findings had been analyzed by quarter end, so the rate describes findings that have reached an outcome. The Q3 cohort reads lower, at 3.8 percent, partly because newer findings have had less time to be reviewed and disputed. We will continue to report the rate as the remainder is resolved.

Reproduction

Reproduction is a separate and deeper step than analysis. Analysis establishes whether a finding is a real vulnerability. Reproduction re-runs the vulnerability against the affected code to establish whether it is exploitable, whether upgrading to the fixed version introduces breaking changes, and what its severity is based on that evidence. VulNow reproduces a prioritized subset of its highest-priority findings: 186 by the end of Q3, up from 56 at the end of Q2.

Severity accuracy

Every VulNow finding is given a severity at detection: Critical, High, Medium, or Low. When the public advisory is published later, it carries its own rating. Severity accuracy compares the two. We count the result in three ways:

- **Same severity:** VulNow and the advisory assign the same rating.
- **Same or neighboring:** the ratings are identical or one level apart, for example VulNow High and advisory Critical.
- **Two or more levels apart:** for example, VulNow Critical and advisory Medium.

For example, CVE-2026-55153 mchange-commons-java: VulNow rated the finding Critical at detection, and the published advisory rated it High. The two ratings are one level apart, so this finding counts as neighboring.

Across all 1,514 confirmed findings with an advisory rating, VulNow's severity at detection was the same as, or one level from, the advisory's rating in **94 percent** of cases. Where the two differ by two or more levels, VulNow's rating is usually the higher one: 76 cases higher, 19 lower. Detection-time ratings lean conservative because they are made before the full evidence is available.

Reproduction narrows the gap further. Table 4 also shows 70 reproduced findings with a published CVE, rated at detection and again after reproduction. These are our highest-priority cases, which is why they start lower than the full set. After reproduction, the exact match rose from 43 to 56 percent, and the remaining differences split evenly between higher and lower. For context, official sources do not fully agree with each other either: on these findings, GitHub and NVD assign the same severity only about half the time. A 56 percent exact match is therefore at the level official scorers reach among themselves.

Comparison	Findings	Same severity	Same or neighboring	Two or more levels apart
All confirmed findings, at detection vs advisory rating	1,514	50%	94%	6%
Reproduced findings with a published CVE, at detection	70	43%	87%	13%
Same 70, after reproduction	70	56%	94%	6%

Table 4. VulNow severity compared with the advisory rating.

What We Cover

The question we are asked most often by prospective customers is which ecosystems and systems VulNow supports. The confirmed set on September 30 answers it directly. It spans **870 distinct open source packages across 12 ecosystems**, up from 97 packages across six ecosystems in Q2.

More than a fifth of confirmed findings sit in projects that are consumed from source rather than from a package registry. These components are frequently vendored or embedded into larger products, which makes them among the hardest for registry-based tooling to track and among the most relevant for manufacturers assessing EU CRA exposure.

Ecosystem	Confirmed findings	Share
npm	520	28%
Maven	305	16%
PyPI	254	14%
Packagist	164	9%
Go modules	93	5%
Linux distributions (Alpine, Debian, RPM)	78	4%
Hex, container images, RubyGems, NuGet	48	3%
Source repository only, no registry package	402	22%

Table 5. Confirmed findings by ecosystem on September 30, 2026.

Findings map to package names and versions, so they correlate directly with software inventory, whether that inventory comes from an SBOM, a software composition analysis tool, or endpoint and build data.

Vulnerability types

30 percent of confirmed findings carry a weakness from MITRE's CWE Top 25 Most Dangerous Software Weaknesses. Denial of service is the largest single class, at 15 percent of confirmed findings. Memory corruption, broken access control, cross-site scripting, and server-side request forgery (SSRF) follow, all CWE Top 25 weaknesses. Types come from the CWEs on the published advisory, not from our own classification.

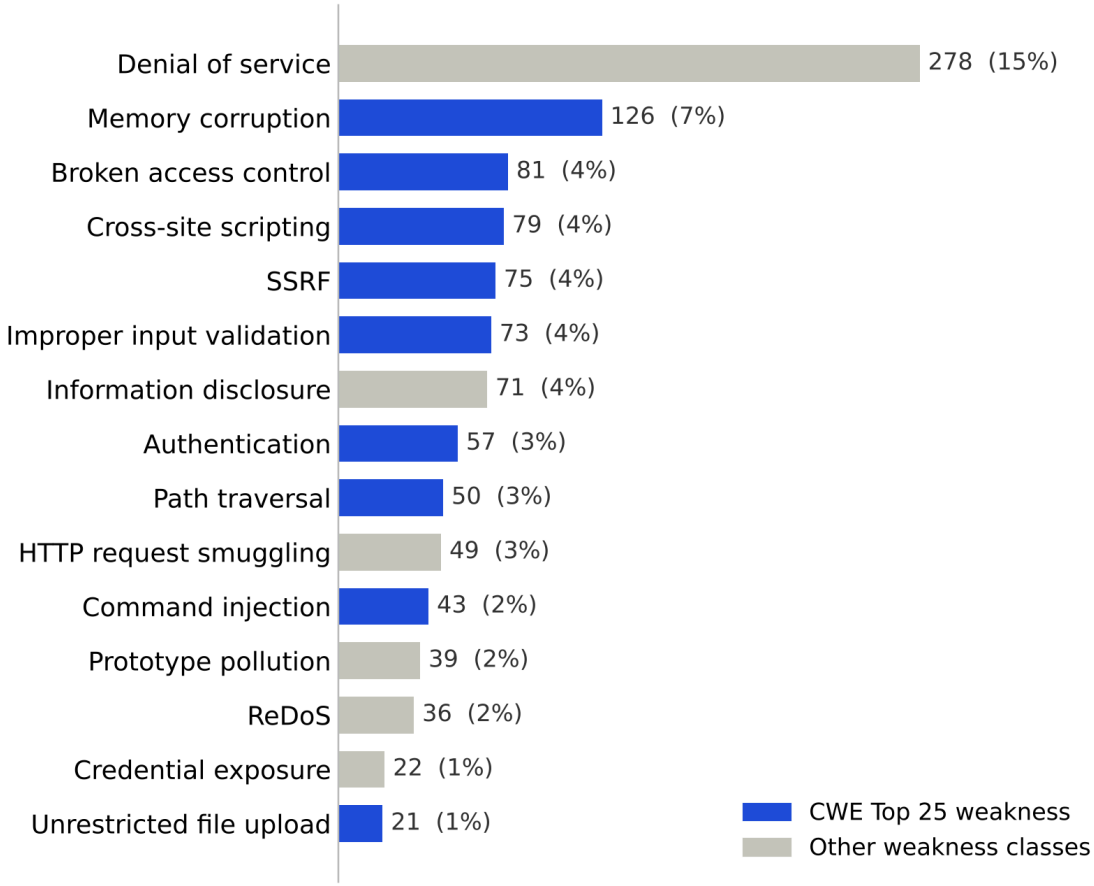


Figure 3. Top weakness classes across 1,864 confirmed findings on September 30, 2026, from the advisory CWEs. A finding with several weakness types counts in each. MITRE CWE Top 25 (2023 edition).

What This Means for a Security Program

Most vulnerability management programs are organized around the published advisory: it starts the clock, sets the severity, and triggers the scanner. The Q3 data suggests three adjustments.

For application security teams

VulNow detects vulnerabilities from the maintainer's own fix, and identifies which released versions of the package contain it. In most cases, then, the upgrade target is already known by the time a finding is raised. A week of advance notice, available for more than half of confirmed findings, is enough to schedule the upgrade, test it, and ship it before the advisory lands and before automated exploitation begins.

For vulnerability management teams

A clean scan does not mean there is no known vulnerability and a KEV listing is not an early warning, when in every Q3 case it trailed our detection by days. Timing after publication matters as well. Tooling that polls feeds or rescans workloads on a fixed cycle adds that cycle to every detection, however current the feed. Event-driven notification, delivered when the signal appears, removes it.

For CISOs

Under the EU CRA, the window before disclosure is now a compliance consideration as well as a security one. Programs that can identify affected components and fixed versions before an advisory exists are in a better position to meet a 24-hour early warning obligation when exploitation is reported. The useful board-level measure is not mean-time-to-patch after a CVE. It is how much of the relevant risk the program could see before the CVE existed.

External sources

- GitHub, "Inside the Advisory Database and what happens when vulnerability volume breaks records," June 29, 2026. https://github.blog/security/supply-chain-security/inside-the-advisory-database-and-what-happens-when-vulnerability-volume-breaks-records/
- GitHub Community, "Multiple published GitHub Security Advisories remain 'No known CVE' after maintainers requested CVEs," discussion #210051, updated October 17, 2026. https://github.com/orgs/community/discussions/210051
- NIST, "NIST Updates NVD Operations to Address Record CVE Growth," April 2026. https://www.nist.gov/news-events/news/2026/04/nist-updates-nvd-operations-address-record-cve-growth
- U.S. Department of Commerce Office of Inspector General, "Evaluation of NIST's Management of the National Vulnerability Database," OIG-26-020-I, 2026. https://www.oversight.gov/reports/evaluation-nists-management-national-vulnerability-database
- European Commission, "Cyber Resilience Act: Reporting obligations." https://digital-strategy.ec.europa.eu/en/policies/cra-reporting
- Regulation (EU) 2024/2847 of the European Parliament and of the Council (Cyber Resilience Act). https://eur-lex.europa.eu/eli/reg/2024/2847/oj
- CISA, Known Exploited Vulnerabilities Catalog and inclusion criteria. https://www.cisa.gov/known-exploited-vulnerabilities
- VulnCheck, Known Exploited Vulnerabilities. https://vulncheck.com/kev
- MITRE, 2023 CWE Top 25 Most Dangerous Software Weaknesses. https://cwe.mitre.org/top25/archive/2023/2023_top25_list.html