// ZERO TRUST CONSULTING | ZTC RESEARCH

INDEPENDENT RESEARCH

THE NEW CYBERSECURITY ASYMMETRY

AI is accelerating vulnerability discovery. Architecture limits viable attack paths; operational response determines how long critical assets remain exposed.

2024–2026 | EVIDENCE · ARCHITECTURE · RESILIENCE

The severity of a vulnerability also depends on its reachability, the privileges it enables, and the boundaries enforced by the architecture.

Report ZTCR–AI–VM–2026–01 | Version 1.0 | October 31, 2026

https://zerotrust.consulting/research

EXECUTIVE OVERVIEW / ZTC RESEARCH

AI is making day-scale response cycles obsolete: critical exposures require containment within hours.

AI-assisted discovery and the automation of exploitation techniques can sharply reduce the time available to protect systems. Deadlines in some frameworks, when carried into internal policies without adapting them to actual exposure, may prove inadequate. ZTC Research calls on standards bodies, regulators, and organizations to review and update their criteria for prioritization, containment, and remediation. Immediate protection should begin within hours; permanent fixes require safe deployment and subsequent verification.

72,833	43 days	≤ 4 h
CVEs disclosed January–September 2026 [01]	**Median full remediation** Known exploited CVEs; 2025 data [29]	**Containment target** Critical exposure; ZTC proposal (§ 9)

Policies become protective when they translate into implemented, tested, and measured controls: reliable asset inventories, segmentation, least privilege, isolation, emergency change procedures, and documented closure. Formal approval is only the beginning.

FROM DISCLOSURE TO IMPACT: SIX STAGES



Stages 01–06: architecture limits access and privileges; patching fixes the defect.

Conceptual model. The sequence illustrates where architecture can intervene: constraining access, privilege, and impact while the underlying defect is corrected.

PRIORITY DECISION

Review and update regulatory criteria and internal policies; demonstrate implementation through trust boundaries, isolation, emergency changes, and technically verified closure.

The research separates documented findings, testable hypotheses, operational proposals, and the limits of available evidence.

RESEARCH OVERVIEW / ZTC RESEARCH

Executive summary

Public data show a marked acceleration in vulnerability disclosures in 2026. In the dataset examined [01], CVEs published from January through September rose from 35,415 in 2025 to 72,833 in 2026 (+105.7%). September totals increased from 4,324 to 14,933 (+245.4% year over year). These figures measure disclosure volume; they do not, by themselves, count AI-discovered flaws or confirmed compromises. [01]

The contribution of AI to vulnerability discovery is supported by verifiable technical cases: autonomous systems have found reproducible defects in large codebases; independent teams identified critical PHP vulnerabilities; and the number of CVEs publicly attributed to AI tools is growing. No consistent global dataset, however, establishes what share of the overall increase can be attributed to AI. [06–10]

Within its own coverage, Google Threat Intelligence Group (GTIG) identified 141 disclosed vulnerabilities with observed active exploitation from January through August 2026, compared with 127 in all of 2025. Active exploitation was observed for approximately 0.23% of disclosures in that sample. This calls for a balanced reading: pressure on security teams is rising, but a published CVE is not an incident. [03] Verizon's 2026 DBIR separately reports a 43-day median to full remediation of KEV vulnerabilities in its 2025 data; that interval is not a global compromise rate. [29]

KEY FINDING
The new asymmetry cannot be established from a single global chart. Within a particular organization, it emerges when relevant findings and potential exploitation speed exceed its capacity to inventory, prioritize, patch, and verify effective closure.

Architecture can restrict access to vulnerable components and reduce privilege, lateral movement, and data exposure after an intrusion. Its effectiveness depends on real configurations and negative testing. It may constrain consequences, but it does not remove the defect or guarantee immunity. [23–24]

Current frameworks set different kinds of obligations. PCI DSS provides a one-month deadline for certain critical updates; CISA BOD 26-04 establishes conditional windows of three, fourteen, or sixty days, or the next major update. For exceptionally high-risk exposures, ZTC Research proposes verified protection within hours and confirmation of permanent remediation within days. [11, 19–22]

Scope of the study. This report examines public sources and architectural principles. It is neither a global census of AI-discovered CVEs nor an experimental evaluation of production systems. Section 12 sets out the limitations and validation criteria.

Conclusions at a glance

- **H1 — Accelerating disclosures:** supported. The sources publish different series but agree on the direction of change. [01–03]
- **H2 — AI as the sole cause of growth:** not established. There are documented AI-attributed cases and samples, but aggregate changes also reflect numbering policy, automation, coverage, and vendors' disclosure cycles. [03, 06, 08]

- **H3 — Remediation saturation:** plausible and measurable. Evaluation requires operational evidence on exposure, applicable findings, time to mitigation, closure capacity, and outstanding debt.
- **H4 — Architectural resilience:** a testable hypothesis for each environment. Isolation, workload identities, egress restrictions, and separated data custody may reduce reachability and blast radius without reducing the global CVE count. [23–24]
- **Eighteen-month horizon:** stress scenario, not a forecast. Jason D. Clinton's October 2026 warning should prompt stronger fundamentals rather than be treated as a confirmed timetable. [15–16]
- **Leadership decision.** Fund and govern complete asset inventories, isolation, contextual prioritization, emergency change paths, validated temporary protection, and evidence-based closure.

Questions guiding the research

The study addresses three questions: How fast are disclosures increasing? How much of that increase can be attributed to AI? And how can an organization reduce exposure even as the number of relevant findings grows?

1. Research question, hypotheses, and method

1.1. Testable hypotheses

H1. CVE disclosures are accelerating. H2. AI contributes to vulnerability discovery, although its aggregate effect has not been quantified. H3. Exposure rises when applicable findings accumulate faster than they can be contained and closed with evidence. H4. Architecture can constrain access paths and impact, provided that isolation, privileges, and egress paths are tested for each asset.

The method combines a critical examination of public sources, comparison of data series, and analysis of technical cases. No independent penetration tests, code audits, active scans, or forensic examinations were conducted. Claims are distinguished as documented facts, source assertions, inferences, or proposed scenarios.

1.2. What each source measures

Source	Unit observed	Application and limitation
CVETodo	CVEs published by month; NVD/CVE/KEV	Primary 2025–2026 quantitative series. Does not universally record how a vulnerability was discovered.
CVE Tools	Monthly CVEs and KEV additions	Corroborates with similar but nonidentical counts. KEV tracks a different category and timestamp.
GTIG	Disclosure, internal risk, and observed exploitation	Threat-intelligence data with its own coverage and criteria—not a universal audit of incidents.
IBM / Unit 42 / Securin	AI-attributed samples and campaigns	Evidence of technical capability; subject to selection bias, differing definitions, and provenance.
FIRST	Disclosure forecasts	Historic projections, not final counts.

Attribution criterion. A vulnerability is considered "AI-discovered" when the method is stated in a primary source or supported by corroborable public attribution. A published CVE does not establish AI discovery; inclusion in KEV likewise does not mean exploitation began during that month.

1.3. Evidence and confidence levels

Level	Criterion	Example
High	Maintainer advisory, explicit data and methods, primary technical records	PHP Group; GTIG for its own dataset; CVETodo for its own series
Moderate	Vendor's own study with verifiable figures but only partial reproducibility	NOVA; IBM sample; Securin tracker
Exploratory	Opinion, social post, forecast, or scenario without external empirical verification	Eighteen-month horizon; saturation models

2. The observed change in vulnerability volume

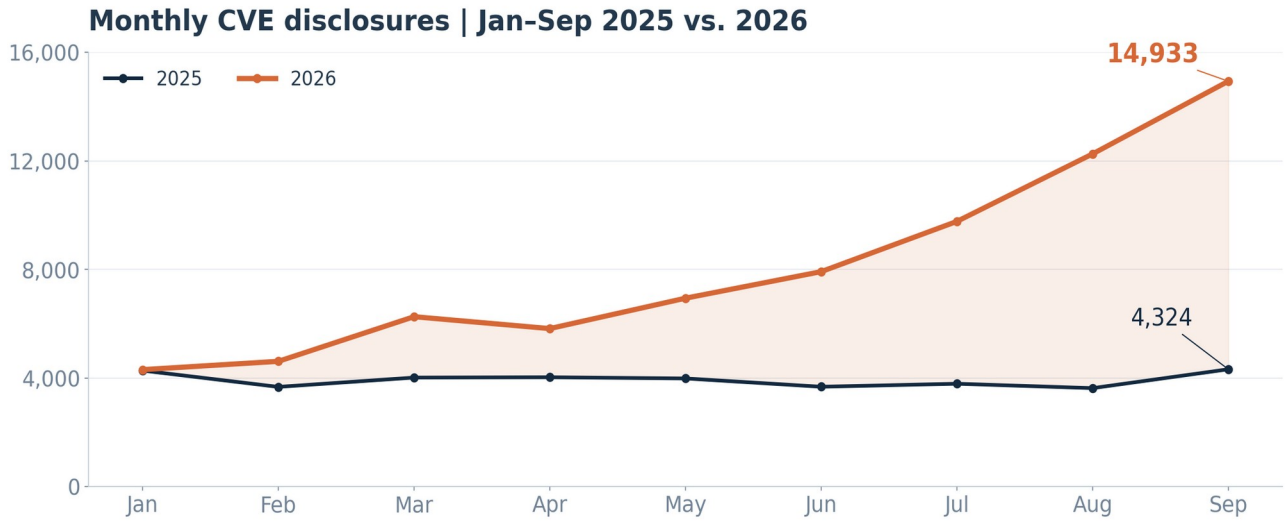


Figure 1. Monthly published CVEs, January–September 2025 and 2026. ZTC Research visualization based on publicly compiled CVE records [01].

From January to September 2026, monthly publications in the series rose from 4,311 to 14,933, a 246.4% increase. Compared with September 2025, the rise was 245.4%. The first nine months totaled 72,833 records, versus 35,415 one year earlier (+105.7%). [01]

The shift unfolded within months, rather than only at year-end. It challenges vulnerability programs organized around isolated quarterly cycles. To translate this trend into risk, each organization must identify which CVEs apply to its assets, which attack paths are viable, and which essential services could be affected.

2.1. Cross-validation and legitimate differences

CVE Tools lists 15,180 CVEs for September 2026 and 4,664 for September 2025, compared with 14,933 and 4,324 in the principal series. Both sources corroborate the direction and magnitude of the increase, but their figures must remain distinct. Differences may reflect query dates, synchronization, or inclusion criteria. [01–02]

Within its monitoring population, GTIG recorded 5,045 disclosures in January 2026 and 10,740 in August. Its series is not directly comparable with the principal dataset. In June, FIRST revised its central annual projection from roughly 59,427 to 66,000 CVEs. These were forecasts at the time, not final counts. [03–05]

2.2. Disclosure volume versus observed exploitation

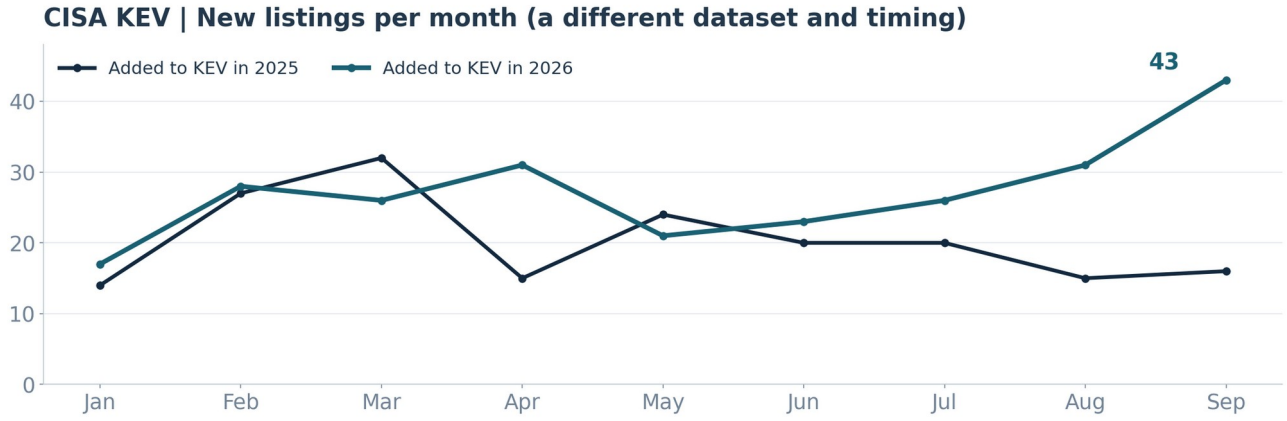


Figure 2. Monthly additions to CISA's KEV catalog according to CVE Tools, January–September 2025 and 2026. Dates refer to KEV inclusion, not initial CVE publication. [02]

Dividing monthly KEV additions by CVEs published that same month and calling it the “true exploitation rate” would be inappropriate: the two streams can concern different populations and dates. The comparison illustrates orders of magnitude and the need for context. [02]

Using a different observation method, GTIG reported 141 vulnerabilities disclosed and actively exploited from January through August 2026, roughly 18 per month; the 2025 monthly average was 10.5. For that 2026 period, approximately 0.23% of disclosed vulnerabilities had observed active exploitation. This finding is limited by GTIG's collection scope and should not be generalized to all vulnerabilities or environments. [03]

2.3. The measured gap between risk and remediation

Verizon's 2026 Data Breach Investigations Report states that, in its 2025 dataset, only 26% of CISA KEV vulnerabilities were fully remediated; the median time to full remediation was 43 days, versus 32 days in the preceding period. These are neither global metrics for every organization nor elapsed times from each CVE's publication. They apply to the population and unit of analysis defined by Verizon. [29]

That lag illustrates the value of architecture: an organization unable to fix a defect immediately must reduce reachability, access, and data exposure while remediation proceeds. Subtracting Verizon's 43 days from a federal three-day mandate would be misleading: populations, clocks, and criteria differ. A sound response measures time to protection, time to closure, and remaining exposed instances separately. [20, 29]

3. The demonstrable role of artificial intelligence

3.1. Demonstrated capability and the limits of extrapolation

NOVA (Unit 42). The team reported examining 3,915 open-source projects and validating 14,090 findings over two months; 99.4% had not previously been reported. Under CVSS v4.0, 39.7% were rated high or critical. These are findings from a research-and-validation campaign, not 14,090 issued CVEs or 14,090 incidents. The study comes from the system's developer; equally extensive external validation of the entire sample has not been published. [07]

IBM X-Force sample. Among 223 vulnerabilities publicly reported as discovered by frontier models, IBM observed 55.2% high/critical severity, 91% remote accessibility, and 25.6% with code-execution potential.

These percentages describe a selected sample; they do not estimate the CVE universe or prove that AI finds only severe flaws. [06]

Securin tracker. It lists 597 CVEs associated with AI actors or systems but distinguishes the strength of attribution: 356 have explicit public evidence of AI-enabled discovery (166 autonomous and 190 human-assisted); another 178 are merely linked to individuals at an AI laboratory, 14 have no disclosed method, and 49 lacked identified public attribution evidence. That distinction matters as much as the headline count. [08]

Verifiable precedent. In 2024, Google Project Zero and DeepMind published a genuine SQLite vulnerability found with Big Sleep; in 2025, CVE-2025-6965 was documented with support from the same program. Technical progress therefore predates the 2026 rise in disclosure records. [13, 17]

3.2. Disclosure growth has more than one driver

CVE counts can also rise because of expansion in the numbering program, changes in assigning authorities, automated records, greater transparency, reviews of legacy code, and large vendor disclosures. GTIG identified about 5,000 CVEs associated with Linux kernel descriptions between January and August 2026, without observing actively exploited zero-days in that subset. [03]

Isolating the causal contribution of AI would require tracking how each flaw was found, removing duplicates, and controlling for changes in coverage, vendors, and disclosure cycles. Public sources do not yet support such an estimate with adequate rigor. The observed increase is verifiable; attributing all of it to AI is not.

4. PHP2Shell: technical evidence and actual scope

On October 21, 2026, PHP announced fixes for CVE-2026-109618 and CVE-2026-109619. In affected contexts, both may allow unauthenticated remote code execution. The first concerns improper memory handling under exhaustion; the second involves internal state retained after memory was freed. The PHP project highlighted greater risk for mod_php and stated that updating is the only complete protection available. [09]

OpenAI reported that Calif identified one flaw with a frontier model, while Wiz independently discovered and disclosed both. PHP maintainers led remediation and coordinated the response with other parties. The evidence documents a real contribution by AI research tools, alongside essential human investigation and software maintenance. [10]

PHP branch	Minimum fixed version	Operational implication
8.2	8.2.35	Plan migration before end of support; confirm the deployed binary.
8.3	8.3.36	Update and verify running processes.
8.4	8.4.27	Update and verify every running instance and container.
8.5	8.5.12	Update; on Windows, validate any subsequent rebuilt releases where applicable.

Scope of exposure. Widespread PHP use does not show that every site runs a vulnerable version or can be exploited. Assessment requires the deployed version, execution mode, configuration, access path, operating controls, and, where appropriate, forensic evidence. [09]

On October 23, PHP also described a specific scenario involving `session.upload_progress.enabled` and dynamic changes to `session.serialize_handler`, tracked as CVE-2026-115560. PHP characterized it as an unusual combination, with a regular-cycle fix pending and configuration mitigation recommended. It should not be confused with the two main CVEs. [09]

Architectural implication. A flaw in the runtime may be triggered before an application performs its ordinary identity checks. API authentication therefore needs to be complemented by isolation, least privilege, segmentation, egress restrictions, and timely updates. When the vendor considers mitigations incomplete, correcting the software remains essential.

5. Eighteen months of pressure: a scenario demanding preparation

5.1. Who is Jason D. Clinton, and why does his assessment matter?

Jason D. Clinton serves as Deputy Chief Information Security Officer at Anthropic and was the company's first senior security leader. He previously led Google Chrome infrastructure security, with experience in ChromeOS, Android Pay, and advanced persistent threats. His career, documented by RSAC Conference, provides context for his warnings; it does not turn a forward-looking opinion into a scientific forecast. [25]

At DEF CON 34 in August 2026, Clinton described the possibility of a particularly demanding 18-to-24-month period. On October 22 he referred to approximately 18 months and emphasized reinforcing foundational controls. He has also advocated isolating critical systems when discovery outpaces patching. These are expert statements, not outputs from a predictive model. [15, 26]

His argument also contains an important tension: systems able to discover bugs might eventually make fixing them cheaper. His expectation that more widely accessible models could reduce the introduction of defects remains a hypothesis rather than an established result. [26]

Clinton's October 22 warning calls for stronger fundamentals before expanding the tool stack. The 18-month period is useful for testing resilience plans, not as an established date for a generalized crisis. [15, 26]

5.2. Three plausible sources of pressure

- **Faster inflow:** research agents expand repository coverage, generate hypotheses, validate cases, and lower the cost of finding difficult defects.
- **Potentially shorter offensive window:** technical disclosures and automation can make attacks easier to reproduce. The effect differs by vulnerability; there is no universal measure of how time to exploitation has changed.
- **Remediation constrained by dependencies:** even an applicable flaw requires ownership, version testing, vendor coordination, change management, and verification of the software actually running in production.

5.3. Illustrative saturation model

Let B_t be the number of applicable findings remaining at the end of month t, λ_t be newly validated inputs, and μ_t be verified closures. The basic accounting identity is $B_t = \max(0, B_{t-1} + \lambda_t - \mu_t)$. It excludes severity, risk, and dependencies, which must be measured separately. It is neither a ZTC forecast nor a market-derived estimate.

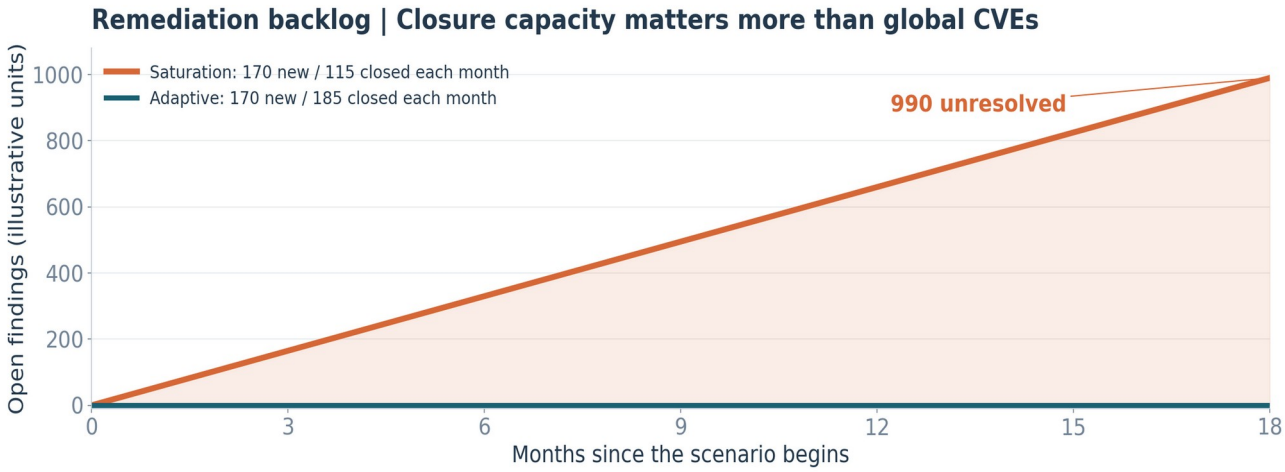


Figure 3. Didactic 18-month simulation using hypothetical—not observed—data. In the saturation case, backlog grows by 55 items per month and reaches 990; sufficient closure capacity prevents accumulation.

The model sharpens the decisions: increase closure capacity, reduce exposure, automate verification, and dismiss inapplicable findings with evidence. Tools that merely multiply alerts can increase outstanding work without improving protection.

5.4. Early warning indicators

Metric	Operational definition	Warning signal
Inventory coverage	Percentage of exposed assets with owner, version, dependencies, and criticality	More assets lack ownership or version information.
Time to triage	Hours from alert to applicability and priority decision	Lengthening even for externally accessible services.
Time to protection	Hours until a temporary control or effective patch is verified.	Exceeds the observed adversary activity window.
Time to verified closure	Hours/days to prove deployed version and removal of exposure	Administrative closure without technical evidence.
Relevant backlog	Applicable open findings on critical assets, weighted by age	New findings exceed closures across several cycles.
Supplier coverage	Services and libraries governed by vendors with verified response timelines.	No proof of update or alternative isolation.

5.5. Extreme cases and trends: keeping metrics distinct

An attack demonstrated in minutes establishes what happened under specific conditions; it does not determine average exploitation time, typical privilege escalation, or the impact on every organization. Interpretation requires the environment, starting point, observed population, time measurement, and success criterion. [29–30]

In the Mandiant case sample, the average interval between initial exploitation and relevant disclosure was -7 days: some flaws were exploited before they became public. That figure does not represent time to compromise for every enterprise. Combining unlike measures into one curve would create false precision. [30]

6. A defensive model focused on reducing exposure

6.1. Vulnerability-level decision sequence

Response should account for applicability, actual reachability, evidence of exploitation, business impact, and change risk. CVSS describes technical severity; EPSS estimates the probability of observed exploitation over the next 30 days under its model; KEV lists vulnerabilities with confirmed exploitation according to CISA. None replaces an assessment of the organization's environment. [11–12]

Stage	Minimum evidence for completion
1. Discover	Verified inventory and version; SBOM or equivalent evidence when applicable.
2. Determine scope	Ingress routes, public exposure, privileges, dependencies, and asset owner.
3. Select response	Recorded KEV/EPSS/CVSS inputs, impact, exceptions, countermeasure, and accountable owner.
4. Contain	Isolation, access/egress restriction, or temporary rule with efficacy and side-effect testing.
5. Remediate	Updated image/package, verifiable deployment, regression tests, and rollback readiness.
6. Verify	Effective-version inspection, repeatable negative test, telemetry, and closure record.

6.2. Controls that narrow the asymmetry

- **Asset and dependency mapping:** identify assets, containers, runtimes, APIs, third-party components, service accounts, deployed versions, and accountable owners; use SBOM and VEX where trustworthy evidence is available.
- **Architectural isolation:** least privilege, separation of administration and data, microsegmentation, private networks, and controlled egress. Verify that secondary integrations do not recreate unauthorized access.
- **Change readiness:** representative environments, progressive deployments, automation, rollback capability, and functional testing. Protection must be verified against the version actually running.
- **Exploitation telemetry:** preserve activity logs capable of revealing attack attempts and confirming compromise; initiate forensics when signs of pre-patch intrusion appear.
- **Intelligence-driven prioritization:** combine KEV, EPSS, reachable attack surface, process criticality, sensitive data, lateral movement potential, and operational dependencies.
- **Third-party accountability:** contractual response times, dependency inventory, maintenance of unsupported components, and evidence of updates.

6.3. Four response decisions

Decision	Indicative triggers	Response
EMERGENCY	Confirmed exploitation and a reachable route to a critical asset; or critical remote RCE with credible activity indicators.	Contain immediately, investigate historical exposure, patch, and verify.
EXPEDITED	High likelihood and real exposure, essential dependency, or severe business impact.	Prioritized deployment against an approved target, with testing.
SCHEDULED	Applicable but with demonstrable constraints and effective compensating controls.	Planned change with expiration, explicit acceptance, and follow-up.
NOT APPLICABLE	Absent product or attack precondition excluded by verifiable evidence.	Document exclusion; reopen if architecture or intelligence changes.

The following states are an internal management proposal; they are not regulatory categories under PCI DSS, ISO 27001, or CISA. Each organization must verify its legal, contractual, and standards obligations by sector, jurisdiction, and date.

6.4. Resilient architecture: limiting harm as CVE volume grows

The number of CVEs is an ecosystem characteristic; effective exposure is an architectural property of each organization. NIST SP 800-207 shifts the focus from trust based on network location to protecting resources with explicit authentication and authorization. NIST SP 800-160 Vol. 2 frames resilience as the ability to anticipate, withstand, recover from, and adapt to adverse conditions—even under compromise. [23–24]

Architectural hypothesis (H4). Given the same vulnerable version, a system with constrained public access, independent identities, least privilege, functional segmentation, and segregated data should expose fewer viable attack paths to critical assets. The hypothesis requires negative tests and telemetry; controls must demonstrate effectiveness in operation. [23–24]

6.4.1. An intrusion chain offers multiple points of interruption

Turning a software defect into actual harm typically requires a chain of conditions: reaching the vulnerable function, triggering it, acquiring usable privileges, reaching additional resources, and executing a harmful action. Independent controls should interrupt different stages. A web application firewall or private network is not sufficient when exploitation originates inside a process that already has authorized access. [23, 24, 27]

6.4.2. Testable controls and safe-failure tests

Layer	Architectural decision	Verifiable evidence
Exposure	Inventory services/dependencies; eliminate public administrative entry points.	Authorized external reconnaissance shows no direct route to consoles or internal runtimes.
Identity	Independent workload identity, ephemeral credentials, per-resource authorization.	Compromising a service account does not grant access to other functions or privileges.

Layer	Architectural decision	Verifiable evidence
Isolation	East–west segmentation, default-deny policies, tenant isolation.	Negative tests across services, networks, and tenants; log denials and bypass attempts.
Execution	Unprivileged processes, sandboxing, read-only filesystems, hardening.	A test RCE remains inside its container without host or secret access.
Data	Minimization, tokenization, and separate custody when appropriate; key-use controls.	Peripheral service cannot read original data or expand authorization.
Egress	Network egress approved by destination, protocol, and purpose; monitor anomalies.	Attempted exfiltration to an unauthorized destination is denied.
Resilience	Immutable deployment, canary/blue-green rollout, tested rollback and recovery.	Drill demonstrates containment, update, and recovery without compromising integrity.

6.4.3. Test scenario: RCE in a public-facing service

Broad-trust architecture: remote code execution in a web service may enable lateral movement when it shares credentials and database access. Segmented architecture: the same defect may compromise the initial process while restricted identities and separated data custody constrain its reach. This is a conceptual comparison, not a test result from a live environment.

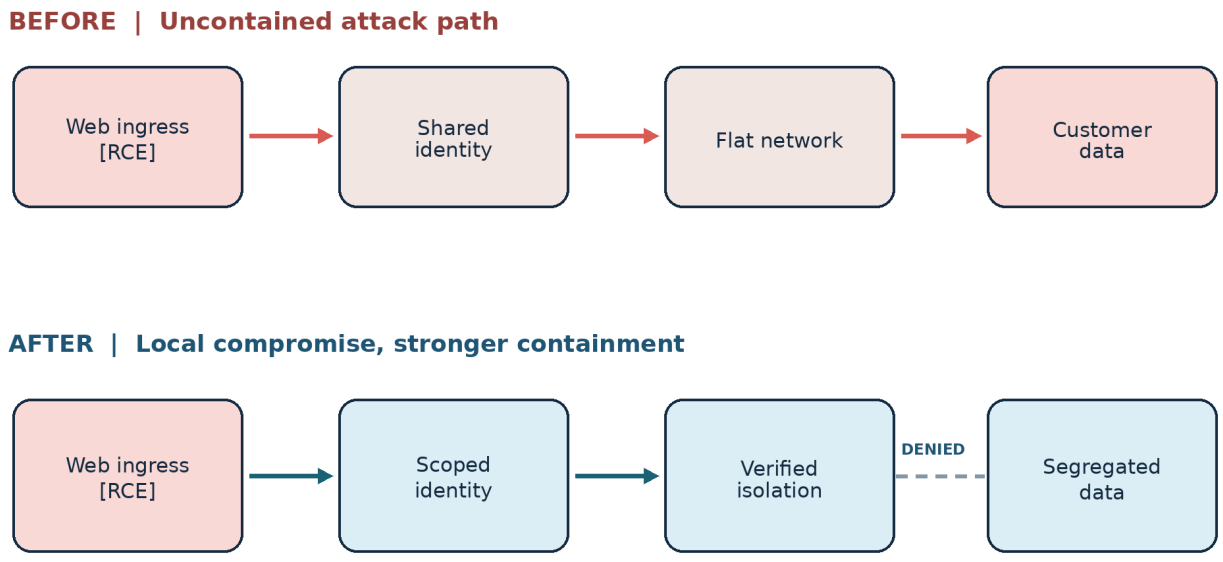


Figure 4. Two topologies facing the same vulnerability. Architecture constrains the reach of compromise; patching corrects the defect.

6.4.4. Measuring effectiveness with verifiable evidence

For each asset, record the affected version, reachability, privileges available after compromise, lateral access paths, data reachable, permitted egress, and time to verified protection. Comparisons should use repeatable

scenarios. A universal percentage reduction in risk would be unsupported without measurements of exploitability and impact.

Useful indicators include negative-test coverage, public assets without direct administration, authorized lateral paths, services sharing identities, hours of essential-asset exposure weighted by criticality, and the 90th percentile of time to containment and closure. Inapplicable CVEs should be documented and retained in the register.

Limits of the model. Compromise of an identity system, exfiltration through permitted channels, configuration errors, or third-party privileges can defeat seemingly strong boundaries. Cryptography and tokenization protect information within their scope, but do not fix memory flaws or replace access controls and patching. [23–24, 27]

6.4.5. Four measurable resilience boundaries

Architectural resilience can be described through four boundaries, each subjected to safe-failure testing. The aim is to verify that selected resources remain inaccessible even when a peripheral component is assumed compromised. [23, 24, 31]

Boundary	Design	Negative test
Ingress	Minimize public entry points; isolate administration.	RCE in an edge service does not open a console or internal plane.
Identity	Workload-scoped privileges, expiration, resource authorization.	Compromised token cannot elevate rights or cross tenant boundaries.
Data	Separate custody, minimization, and usage restrictions.	Compromised web service cannot retrieve original data beyond its need.
Egress and recovery	Explicit outbound rules, rollback, evidence-backed restoration.	Unauthorized exfiltration blocked; change is reversible and verified.

Tests should cover genuine service credentials with insufficient permissions, service-to-service movement, tenant isolation, unauthorized egress, and deployment failures. A firewall, a network diagram, or the presence of a token does not by itself establish that these boundaries work. [23–24, 31]

7. Implications for financial services and critical suppliers

In banking, payments, and platforms holding personal data, vulnerability risk depends on the trust boundaries that can be crossed. The relevant questions are whether a flaw compromises identities, reaches sensitive information, or leverages standing third-party privileges. A rigorous assessment reconstructs these full paths rather than merely marking a CVE as present.

- **Public-facing plane:** API gateways, portals, reverse proxies, runtimes, and exposed libraries; distinguish application-layer exploitation from compromise of the execution environment.
- **Identity plane:** ephemeral credentials, transaction-specific authorization, phishing-resistant MFA where appropriate, workload permissions, and separation of duties.
- **Data plane:** limit PAN/PII and other sensitive-data exposure, minimize copies, use tokenization where appropriate, and test read and egress restrictions.
- **Supplier plane:** record critical libraries, patching responsibilities, disclosure agreements, and trust boundaries in integrations.

- **Resilience plane:** negative isolation testing, operational rollback, verified restoration, retesting, and analysis of exploitation logs before and after closure.

Compliance perspective. Conformity with a standard does not establish protection against future vulnerabilities. Similarly, a list of scan findings cannot replace validation of control scope and effectiveness. Business continuity requires vulnerability management, architecture, and operations to work together.

7.1. Governance questions that should have documented answers

- How many exposed services currently lack an accountable owner, an identified runtime version, or documented dependencies?
- Which new CVEs affect our assets, and which have an attack path and sufficient privileges to cause business impact?
- What are the median and 90th-percentile times from alert to isolation, deployed fix, and technically verified closure?
- What happens when an essential supplier has a critical vulnerability but cannot deliver a patch for several days?
- What tests demonstrate that customer, service, and network boundaries survive compromise of a peripheral component?

8. Current vulnerability-management and patching timelines

Current frameworks impose different types of requirements and guidance: some set maximum deadlines by severity; others define assessment frequencies or asset-specific conditions. Scan frequency, remediation deadline, and technical recommendation are different concepts. Keeping them distinct avoids incorrect regulatory conclusions. [11, 19–22]

8.1. Comparison of current requirements and guidance

Framework	Documented deadline or frequency	Correct interpretation
PCI DSS 4.0.1	Req. 6.3.3: patches/updates for critical vulnerabilities within one month of release. Others are governed by entity-defined risk.	Binding for in-scope PCI systems; does not mean waiting the full month. [19]
CISA BOD 26-04	Conditional maximums of 3, 14, or 60 calendar days, or the next major update; the 3-day tier may require forensic triage. Four asset-level factors.	Binding for covered U.S. federal agencies. Full implementation phase begins Dec. 7, 2026. [11, 20]
CIS Controls v8.1	7.2: review remediation strategy monthly or more often; 7.3/7.4: automated patching monthly or more often; 7.7: remediation monthly or more often.	Recommended process frequencies, not a universal grace period for every CVE. [21]
DORA / RTS 2024/1774	Art. 10: scan at least weekly for assets supporting critical/important functions; requires emergency procedures and defined patch-installation and escalation times.	Article 10 does not set a universal 3-, 7-, or 30-day SLA for every flaw. [22]

Framework	Documented deadline or frequency	Correct interpretation
NIST SP 800-40 Rev. 4	Treat patching as preventive maintenance: identify, prioritize, acquire, install, verify; establish an enterprise risk-based strategy.	Technical guidance; no universal per-CVE deadline. [28]
ISO/IEC 27001:2022 + ISO/IEC 27002:2022	Annex A control 8.8: obtain information on technical vulnerabilities, assess exposure, and take appropriate measures. The standard sets no universal patching deadline in days.	Risk management and demonstrable treatment under the ISMS; the organization must justify priorities and time frames. ISO 27002 provides implementation guidance. [33–34]

How deadlines are measured. Under PCI DSS, the one-month period for critical updates begins when an applicable patch is published. Under CISA BOD 26-04, the clock follows the conditions established by the directive, including KEV listing or detection and reporting of a flaw in a federal asset. A change in exposure or threat intelligence may require earlier action. [19, 20]

ISO/IEC 27001 and 27002. Technical vulnerability management (A.8.8) relates to threat intelligence (A.5.7), configuration (A.8.9), change management (A.8.32), and ICT readiness for business continuity (A.5.30). ISO/IEC 27001 requires risk treatment but does not set a universal number of days for every patch. Internal timelines must be justified by exposure, criticality, and response capability. [33–34]

8.2. How frameworks should evolve amid AI-assisted discovery

First proposal: set targets according to confirmed exploitation, feasible automation, public exposure, available privileges, and potential harm to data and essential services, as well as severity. Each criterion should be verifiable and auditable.

Second proposal: track two intervals. One runs from a credible alert to effective protection; the other ends with a verified patch or permanent removal of the defect. A temporary measure should leave the case open until correction is complete.

Third proposal: regularly test service and tenant isolation, identity safeguards, and egress restrictions. Fourth: define emergency change paths, rollback arrangements, and verifiable response obligations for critical suppliers. [23–24, 27]

ZTC Research calls on standards bodies to revisit their criteria and organizations to update the internal policies that implement them. Risk assessment must consider exposure, confirmed exploitation, attack automation, and potential effects on identity and data. CISA BOD 26-04 illustrates contextual prioritization; this report additionally proposes verified containment within hours, isolation testing, and residual-risk reporting. These are technical recommendations, not current mandates. [20, 32]

9. ZTC Research proposal: protection within hours, verified fixes within days

The following targets are a voluntary proposal for significant assets. They should be embedded in approved internal policies and implemented as tested, measurable controls, with named owners, evidence, and time-bound exceptions. They do not replace applicable legal, contractual, or standards requirements; any stricter binding deadline prevails. Containing an attack path is not the same as eliminating a vulnerability.

9.1. Risk-based proposed thresholds

Tier	Contextual trigger	Proposed internal target
A / Emergency	Applicable active exploitation; reachable KEV; automatable remote RCE with high impact and external attack path.	Containment ≤ 4 h; immediate forensic triage; target closure ≤ 72 h if a safe fix exists.
B / High exposure	Proven external path plus plausible exploitation / working exploit / critical service.	Protect ≤ 24 h; target closure ≤ 7 days.
C / Exposed, not exploited	Applicable high/critical vulnerability, public access, verified defenses, and no confirmed activity.	Protect ≤ 72 h; target correction ≤ 14 days.
D / Scoped internal exposure	Not public or reachable from untrusted zones; limited scope documented and verified.	Review ≤ 7 days; target correction ≤ 30 days.
E / Low relevance	Noncritical, not exploited, documented inventory and paths, no changing indicators.	Risk-based change window; continuous review, not silent closure.

At Level A, if no patch exists or immediate deployment would jeopardize continuity, the organization should consider removing public exposure, restricting use, or isolating the affected system with evidence of effectiveness. Any exception requires immediate escalation, continuity safeguards, and preservation of forensic evidence. The case remains open until permanent remediation. [20, 23–24]

9.2. Auditable metrics

Time to protection (MTP): hours from a credible alert to evidence that a control interrupts the attack path. Time to closure (MTC): hours or days until the permanent fix is verified. Essential-asset exposure (EAEC): accumulated hours of vulnerable assets with relevant attack paths, weighted by criticality. Boundary coverage: the proportion of negative tests passed for identity, network, data, tenant, and egress boundaries.

Metrics should be reported by severity and percentile, with populations and exclusions stated. CVE totals alone cannot represent operational workload: one flaw present in hundreds of systems can create greater exposure than hundreds of CVEs that do not apply to the inventory.

9.3. Cost and continuity constraints

Reducing exposure time also requires investment in automated testing, safe deployment, response capacity, redundancy, and reversibility. Faster patching without validation can cause outages. Economic analysis should compare exposure hours, control costs, intervention frequency, and continuity outcomes without counting hypothetical benefits as realized savings.

9.4. Investment priority and accountability

Investment should start with architectural boundaries that constrain impact, then expand automated updates and testing, and mature through sustainable observability and response. Every initiative needs an owner, total cost, implementation horizon, residual-risk statement, and evidence of effectiveness. A reduction in expected incidents is a potential benefit, not a financial saving already realized.

Before enforcing shorter deployment windows, organizations should test isolation, rollback, restoration, and integrity checks. Faster response should improve resilience without causing disproportionate disruption.

9.5. Three clocks and one closure criterion

Clock 1, recognition: from a credible alert to a recorded decision on applicability. Clock 2, protection: until a control demonstrably interrupts the attack path. Clock 3, closure: until a fix is deployed and validated on affected assets, with functional and security tests.

When a vulnerability is exploited before disclosure, an internal clock does not capture the full historical exposure. Retrospective event review, signs of earlier intrusion, and log-retention limits must also be considered. Administrative closure without technical proof can conceal residual compromise. [29–30]

10. Alternative interpretations and scope of conclusions

First objection: more CVEs do not necessarily mean worse software. Broader review can reveal old defects without increasing the rate of newly introduced flaws. Greater visibility can favor defense, while adding pressure on remediation capacity. [03–05]

Second objection: AI can also strengthen defense. Coordinated work on PHP and Big Sleep shows how discovering and reporting flaws before exploitation can give defenders additional time. The benefit depends on fixes reaching deployed systems. [10, 13, 17]

Third objection: some counts may overstate the problem. Distinguish a technical finding, a CVE, an affected instance, a working exploit, an attack attempt, and a confirmed compromise. NOVA results and AI-attributed samples must not be added to global totals as independent new CVEs. [06–08]

Fourth objection: the 18-month interval is a subjective estimate. It is useful for resilience exercises, not an established date of crisis or recovery. The report does not turn that judgment into a statistical forecast. [15–16]

Fifth objection: observed exploitation represents a small fraction. That fraction can still produce serious harm when widely deployed platforms are affected. Conversely, many inapplicable CVEs may create little operational exposure for an organization. Risk arises from the combination of criticality, reachability, and response speed. [03, 18]

11. Action roadmap: first 72 hours, 14 days, and continuous improvement

Phase	Objective	Evidence of outcome
0–4 hours	Assign an on-call team, locate externally accessible critical assets, and activate emergency containment.	Recorded decision, owners, and testable isolation path.
4–24 hours	Confirm applicable CVEs and attack paths; apply temporary protection and monitor activity.	Scope record, blocking tests, and review for previous compromise.
24–72 hours	Deploy safe fixes, retest, roll out progressively, and prepare rollback; verify lateral access.	Running versions, negative tests, and closure of highest-risk cases.

Phase	Objective	Evidence of outcome
Days 4–7	Extend least privilege, dependency inventory, and egress restrictions across essential systems.	Route/owner matrix; service and tenant isolation tests.
Days 8–14	Rehearse controlled exploitation and recovery; formalize risk-based targets.	Verification coverage, p90 protection time, time-limited exceptions.
Days 15–30	Automate the patch-and-test chain; agree response terms with suppliers.	Input/closure balance and evidence of declining exposure.
Ongoing: months 1–18	Sustain redesign around trust boundaries; monitor trends and verify independently.	Contained compromise scenarios and recurring resilience evidence.

Short-cycle governance. During the first fourteen days, alerts affecting essential assets should be reviewed daily. Thereafter, continuous monitoring should be paired with weekly reporting on exceptions and trends. Leadership should track at least four measures: critical assets without coverage, high-impact findings still exposed, the 90th percentile of time to protection, and the relationship between new cases and verified closures. Defensive AI investment is justified by measurable outcomes, not the number of alerts generated.

12. Scope and limitations of the research

This is a documentary, comparative, and architectural study. It distinguishes public observations, reasoned inferences, and proposed measures requiring validation in specific environments. Stating these boundaries makes the analysis reproducible and identifies what evidence would support or refute each hypothesis.

12.1. Methodological limitations and validity safeguards

Dimension	Limitation	Validity safeguard
CVE and KEV series	Repositories differ in ingestion, timestamps, retroactive changes, and inclusion criteria. Publication is not the same as discovery.	Compare trends within each source and state the cut-off; do not create an artificial single global total or KEV/CVE exploitation rate. [01–05]
AI attribution	There is no universal verifiable field indicating how each CVE was discovered. Vendor-reported samples may reflect selection bias.	Separate documented cases from aggregate series. Do not estimate AI's causal global share. [06–10]
Real-world exploitation	CISA KEV and security-vendor telemetry cover different populations; absence of observation is not proof of no attacks.	Distinguish observed exploitation, exploit availability, attempts, and compromise. State each claim's denominator. [02–03]
Remediation time	Frameworks specify requirements or frequencies for different populations; field timings use non-comparable clocks, assets, and samples.	Separate recognition, protection, and closure; do not confuse observed medians with universal regulatory deadlines. [19–22, 29]
Architectural effectiveness	The trust-boundary model rests on technical principles; no active tests or controlled comparisons were performed on customer infrastructure.	Provide mechanisms and proposed negative tests without claiming numerical incident reduction or immunity. [23–24, 31]

Dimension	Limitation	Validity safeguard
Sector and economic scope	International sources are not representative samples of Mexico or Latin America. Entity-level costs and impacts were not measured.	Avoid extrapolating sector rates and financial returns; propose locally testable exposure, continuity, and cost metrics.
Horizon and currency	The eighteen-month warning is not a statistical prediction. AI capabilities and records may change after the 2026 cut-off.	Use the horizon as a stress scenario, not a firm date; reconfirm figures in future editions. [15–16]

12.2. What the findings support

The evidence supports three conclusions: disclosures increased in the series examined; relevant findings have been made with AI assistance; and architecture with tested trust boundaries can restrict attack paths. The magnitude of this effect depends on assets, configuration, and sustained enforcement.

Available data do not support attributing all global CVE growth to AI, treating every record as exploitable, imposing one response deadline on every organization, or assigning universal percentages to architectural risk reduction. The response targets require testing; the simulations do not describe results achieved in live systems.

12.3. Further evidence required

- **Temporal reproducibility:** publish dated snapshots and queries for each series, deduplication rules, and a record of subsequent corrections.
- **Independent attribution:** record the specific role of AI, human review, proof of concept, and disclosure timing for each finding.
- **Architectural validation:** in an authorized laboratory with synthetic data, compare the same vulnerable component under different designs; measure reachability, escalation, cross-tenant access, and egress.
- **Longitudinal monitoring:** track the three clocks (recognition, protection, closure) for each asset, including percentiles, containment failures, availability effects, and verifiable costs.

12.4. Revision and potential falsification criteria

The architectural hypothesis must be revised if a controlled test shows that declared boundaries can be crossed, unauthorized data accessed, or unapproved egress maintained after one component is compromised. Response objectives also warrant adjustment when containment cannot be verified or causes greater operational harm than the risk reduced. Such falsification conditions are part of the method.

12.5. Hypothesis status and the next decisive test

Hypothesis	State of evidence	Next decisive test
H1 · CVEs	Growth observed in the examined series.	Recreate database snapshots and isolate effects of recording changes. [01–05]
H2 · AI	Real documented cases; global causal effect unquantified.	Build a cohort with verified discovery-method attribution and independent comparison. [06–10]

Hypothesis	State of evidence	Next decisive test
H3 · Response	Plausible mismatch; size depends on environment.	Measure new findings, definitive closure, time to protection, and backlog per essential asset. [29]
H4 · Architecture	Technically supported mechanism; empirical effect not measured here.	Test the same flaw under two designs and document blocked and residual paths. [23–24, 31]

13. Conclusions

First. The increase in vulnerability disclosures during 2026 is verifiable and calls for reassessing monthly or quarterly management cycles. Counts vary by source and date; the acceleration trend is consistent. [01–03]

Second. AI has already contributed to discovering significant defects, including older flaws in widely deployed software. PHP2Shell and NOVA illustrate this capability, although their results do not represent the full set of disclosed CVEs. [07, 09–10]

Third. Rising disclosure counts alone neither establish AI as the sole cause nor demonstrate a proportional increase in successful attacks. Disclosure, severity, exploitation likelihood, and harm require separate measures. [03, 06, 08]

Fourth. Architecture is a structural defense: it can reduce reachability, privilege, and the scope of compromise even when a flaw remains. Its effectiveness must be tested at trust boundaries, since the same vulnerable version can pose different risks under different designs. [23–24]

Fifth. Standards bodies and organizations should review and update criteria and policies to meet the speed of AI-assisted threats. Improvement depends on implemented, tested, and measured controls; the central metric is time to effective protection, not the number of patches reported.

Sixth. The next 18 months are a plausible stress scenario, not a definite date for crisis. Under active exploitation and critical exposure, containment should work within hours and permanent fixes should be verified as quickly as safely possible. PCI DSS and CISA deadlines have distinct scopes and conditions; ISO/IEC 27001 requires risk treatment without prescribing a universal number of patching days. [11, 19–20, 33–34]

CORE FINDING · ZTC RESEARCH

Vulnerability disclosures have accelerated, and AI systems are expanding the ability to identify consequential defects. The most useful measure of resilience is each organization's verified exposure relative to its actual capacity to detect, isolate, remediate, and validate attack paths to critical assets.

Central implication. Resilience is demonstrated when an organization identifies its attack paths, contains critical access, constrains privileges, corrects defects, and validates recovery with evidence.

Appendix A. Data underlying the figures

Month	CVE 2025	CVE 2026	YoY change	KEV 2025	KEV 2026
Jan	4,276	4,311	+0.8%	14	17
Feb	3,676	4,617	+25.6%	27	28
Mar	4,019	6,261	+55.8%	32	26
Apr	4,030	5,823	+44.5%	15	31
May	3,987	6,942	+74.1%	24	21
Jun	3,681	7,920	+115.2%	20	23
Jul	3,792	9,770	+157.6%	20	26
Aug	3,630	12,256	+237.6%	15	31
Sep	4,324	14,933	+245.4%	16	43

The CVE and KEV series come from sources with different dates and criteria [01–02]. They are presented to compare trends without manufacturing an exploitation rate from the two datasets. Counts may change with retroactive updates or synchronization; CVE disclosure dates and KEV inclusion dates need not coincide.

Appendix B. Operational glossary

Term	Working definition
CVE	Identifier for a public vulnerability; neither an incident nor proof that a specific asset is exposed.
CWE	Software weakness category that groups classes of defects.
KEV	Catalog of vulnerabilities with exploitation confirmed in the wild under CISA's criteria.
EPSS	Statistical estimate of observed exploitation probability over the next 30 days; does not calculate business impact.
CVD	Coordinated vulnerability disclosure to allow fixes and protection before broad public exposure.
RCE	Remote code execution; a potential effect whose context, prerequisites, and privileges must be assessed.
SBOM / VEX	Software Bill of Materials / documented statement about affected or unaffected status, supported by evidence.
Verified closure	Technical confirmation that a measure reduces or removes exposure on the system actually deployed.

Sources and traceability

Document cutoff: October 31, 2026. The references allow readers to trace published claims. The report synthesizes sources and offers independent analysis; it does not include results from proprietary active testing.

[01] CVETodo — CVE Statistics 2026; monthly series for 2025–2026. Accessed October 31, 2026. View source

[02] CVE Tools — Monthly CVE trends; snapshots and additions to CISA KEV. Accessed October 31, 2026. View source

[03] Google Threat Intelligence Group — Vulnerability Discovery and Exploitation Trends in the AI Era, September 30, 2026. View source

[04] FIRST — 2026 Vulnerability Forecast, February 11, 2026. View source

[05] FIRST — Mid-Year Vulnerability Forecast, June 15, 2026. View source

[06] IBM X-Force — AI Can Find Serious Vulnerabilities at Unprecedented Speed. Can Defenders Keep Up?, October 2026. View source

[07] Palo Alto Networks Unit 42 — The Frontier AI Vulnerability Burst: Industrializing Autonomous Zero-Day Discovery, August 4, 2026. View source

[08] Securin — AI-Discovered CVE Tracker; attribution database with evidence levels, October 27, 2026. View source

[09] PHP Group — Security advisories and fixed PHP2Shell versions, October 21 and 23, 2026. View source

[10] OpenAI — Patch the Planet: Global Response Coordination as Agents Uncover Critical Vulnerabilities, October 21, 2026. View source

[11] CISA — BOD 26-04: Prioritizing Security Updates Based on Risk, June 2026. View source

[12] FIRST — EPSS Frequently Asked Questions: scope, applications, and limitations. View source

[13] Google Project Zero — From Naptime to Big Sleep, November 1, 2024. View source

[14] Jon Hernández — "La IA está rompiendo la seguridad de Internet" [Spanish-language program recap], October 26, 2026. View source

[15] Jason D. Clinton — Post on X, October 22, 2026; content corroborated through secondary reproduction. View source

[16] Mantic — Black October and the Future of Cybersecurity, October 27, 2026. View source

[17] Google — Big Sleep and the discovery of CVE-2025-6965, 2025. View source

[18] CISA — Vulnerability Review: baseline controls, prioritization, and exploitation, 2026. View source

[19] **PCI Security Standards Council —** FAQ 1597, distinguishing requirements 6.3.3 and 11.3; updated May 2025. View source

[20] **CISA —** BOD 26-04, official directive, June 10, 2026; conditional deadlines. View source

[21] **Center for Internet Security —** CIS Controls v8.1, Control 7; process and remediation frequencies. View source

[22] **European Commission —** Delegated Regulation (EU) 2024/1774, Article 10 (DORA). View source

[23] **NIST —** SP 800-207, Zero Trust Architecture, 2020. View source

[24] **NIST —** SP 800-160 Vol. 2 Rev. 1, Cyber Resiliency Engineering, December 2021. View source

[25] **RSAC Conference —** Jason Clinton, professional biography and background. View source

[26] **Clarín / DEF CON 34 —** Jason Clinton's 18–24-month warning, August 2026. View source

[27] **Microsoft Learn —** Zero Trust, trust boundaries, and defense in depth: architecture guidance. View source

[28] **NIST —** SP 800-40 Rev. 4, Enterprise Patch Management Planning, April 2022. View source

[29] **Verizon —** 2026 Data Breach Investigations Report; KEV coverage and median remediation time (2025 data). View source

[30] **Google Cloud / Mandiant —** M-Trends 2026; estimated -7-day time to exploitation in the sample. View source

[31] **NIST —** SP 800-207A; workload identity and service-level access controls in cloud-native systems. View source

[32] **FedRAMP —** Institutional response to CISA BOD 26-04 and implementation timetable. View source

[33] ISO — ISO/IEC 27001:2022, Information Security Management Systems — Requirements. View source

[34] ISO — ISO/IEC 27002:2022, Information Security Controls — implementation guidance. View source