A Deterministic PAIN and Remediation Method for FedRAMP Compliance and Security-Benchmark Findings

Matthew Venne
Chief Technology Officer, stackArmor

July 2026

Abstract

FedRAMP's Vulnerability Detection and Response (VDR) and Vulnerability Evaluation and Reporting (VER) rules treat a vulnerability as any weakness — not only a CVE but a misconfiguration, an exposure, or a control gap. Security-benchmark scanners (CIS, DISA STIG, cloud-configuration checks) therefore emit FedRAMP vulnerabilities, yet they rarely emit the inputs the core CVSS-Environmental method needs: a benchmark check reports a rule identifier and a Pass/Fail, seldom a C/I/A impact vector, and often no severity at all. This memo supplies a deterministic classifier for that data. A severity-by-asset effect matrix (mirroring the CVE method's impact-by-requirements combination) produces a customer-effect word; FedRAMP's effect-and-scope table turns it into a PAIN level; a finding-level *internet-exercisability* test selects the remediation column; and the same pinned `VDR-TFR-PVR` matrix used by the core memo selects the deadline. The method consumes only what scanners and ordinary asset inventory already provide, fails safe and loud on provider-controlled unknowns, and discloses its one calibrated default rather than hiding it. A back-test on a real, sanitized CIS Google Cloud Platform scan of 1,378 findings shows no incident-class flood: fourteen findings (1.0%) ride the fast clock. This is an addendum to *A Deterministic, CVSS–Environmental Method for FedRAMP VDR/VER Vulnerability Prioritization*; the two share notation, the asset model, and the deadline matrix, and are meant to be read as one family.

Contents

1 Status of this memo and terminology

This document is informational. It does not define a FedRAMP requirement; it specifies a standardized *method* for satisfying the existing VDR/VER requirements when the finding under evaluation is a security-benchmark or compliance result rather than a CVE.[1] It is an addendum to the CVSS-Environmental memo (hereafter "the CVE memo"); it reuses that memo's asset model, its remediation matrix, and its treatment of dispositions, and cites it rather than restating it.

The key words MUST, MUST NOT, SHOULD, SHOULD NOT, and MAY are to be interpreted as described in RFC 2119.

PAIN
Potential Agency Impact, the FedRAMP N1–N5 rating of a finding.

Grade
This method's governed severity of a benchmark finding: Major, Moderate, or Minor.

Customer effect
The FedRAMP effect word a finding lands on: Minimal, Narrow, Disruptive, or Debilitating.

CR/IR/AR
The CVSS Environmental Confidentiality / Integrity / Availability *Requirements* of the affected asset after the intended-use ceiling is applied; the profile-derived band summarizes them for this addendum.

LEV / IRV
Likely Exploitable / Internet-Reachable Vulnerability, the two VER axes that select the remediation column.

Internet-exercisable
This method's operational test for whether an unauthenticated internet actor can exercise a benchmark finding as it stands. It is defined in Section 7 and is *not* a redefinition of IRV.

Class
The provider Certification Class (A–D) selecting a remediation table.

[1] The VDR and VER rules launched on 2026-06-24 as part of the FedRAMP Consolidated Rules for 2026 and apply to both offerings of every FedRAMP certification type.

Three terms are held strictly apart throughout and are never interchangeable. *Internet-accessible* names an entry point — a listener an outside actor can reach directly. *Internet-reachable* is FedRAMP's broader status: a finding that might be triggered by a payload originating on the public internet, including by an indirect path with no direct route. *Internet-exercisable* is this method's own operational test for benchmark findings, defined below; it is a deliberately conservative proxy, not a synonym for either of the other two.

2 Scope: benchmark findings are FedRAMP vulnerabilities

IN PLAIN TERMS

A vulnerability, to FedRAMP, is any weakness an attacker could use — not just a numbered CVE. A misconfigured storage bucket, a disabled log, an open management port, a missing encryption setting: each is a vulnerability the rules expect a provider to detect, evaluate, and remediate on a clock. The trouble is that the scanners that find these things speak a different dialect from CVE scanners. They tell you which rule failed and on what resource, but usually not "how bad" in the CVE sense and often not even a severity. This memo builds the missing bridge from that thin data to a defensible PAIN level and deadline.

FedRAMP's definition of a vulnerability is deliberately broad: it covers control gaps, misconfigurations, exposures, and other weaknesses, not only CVEs.[2] A security-benchmark failure is therefore a FedRAMP vulnerability, and the VDR/VER duties — estimate its Potential Agency Impact, evaluate exploitability and reachability, and remediate within the timeframe matrix — apply to it in full. The CVE memo answers those duties for findings that arrive with a CVSS vector. Benchmark findings do not, and that gap is what this memo closes.

The obstacle is the input data, not the output vocabulary. A CVE scanner emits an impact vector the CVSS-Environmental formula can consume directly. A benchmark scanner emits a rule or category identifier and a Pass/Fail state. Some add a severity of their own, but those labels are not uniform across tools. CIS Benchmarks publish no severity at all: the Level 1 and Level 2 profiles are hardening *scopes*, not severities, and MUST NOT be read as one.[3] DISA STIG content is the exception that proves the rule: it carries a severity category (CAT I/II/III) that scanners translate faithfully. So the method must work from a Pass/Fail and a category name, respect a governed severity when one genuinely exists, and never invent one where none does.

Two design commitments follow, and both are load-bearing. First, the method reuses the asset metadata the CVE method already requires — the affected asset's independently dimensional security-impact profile, its multi-agency scope, and the offering's Certification Class — and asks for no benchmark-specific metadata that ordinary scanners and inventory do not already provide. The profile may be assigned directly or derived from an archetype or another governed mapping. The only governed artifacts this addendum adds are a severity source registry and a small per-benchmark determination table, both described below. Second, exploitability and reachability stay on the axes FedRAMP built for them: they select the remediation *column*, never the PAIN level. A single fact is never counted twice.

The normative FedRAMP anchors for what follows are the VER estimate-impact clause,[4] the

[2] FedRAMP, *Definitions* (*vulnerability*). Pinned permalink: FedRAMP/2026-markdown@2a75592 (commit of 2026-06-24); accessed 2026-07-16.

[3] CIS, *CIS Benchmarks FAQ* (Level 1 and Level 2 profiles). cisecurity.org/cis-benchmarks-faq; accessed 2026-07-16.

[4] FedRAMP, *VER* (*Estimate Potential Agency Impact*): FedRAMP/2026-markdown@2a75592.

VER evaluation factors,[5] the VDR mitigation-and-remediation expectations,[6] and the LEV definition,[7] each pinned at commit 2a75592 (2026-06-24).

3 Disposition is an overlay, not a gate

IN PLAIN TERMS

A scanner finding can receive a PAIN score as soon as it is detected. Internal analysis may later show that the result is a false positive, does not apply to the asset's role, or is already neutralized by another control. That later decision is the disposition. PAIN says how consequential the reported condition would be if it applies; disposition says whether and how it applies in the running environment. They are separate decisions, and disposition is not a precondition for scoring.

A scanner-reported `Fail` may therefore enter the PAIN calculation before internal analysis is complete. Its computed PAIN and derivation remain on the audit record even if a later disposition moves the finding out of the active remediation queue. Keeping the score makes the decision auditable and reversible, and it allows PAIN to help prioritize which findings receive analysis first. The possible outcomes remain distinct:

- A `Pass` is not a finding.
- A result that does not match the effective state is a **false positive**; its disposition and evidence are recorded alongside the computed PAIN.
- A recommendation that does not apply to the asset's actual role is **not applicable**. An N/A disposition is an auditable out-of-scope decision and carries a mandatory reportable rationale — it is not a silent drop, and it does not erase the original score.
- A failed configuration neutralized by an effective control is **fully mitigated**; the evidence and the retained PAIN are recorded, exactly as in the CVE memo's disposition overlay.
- A weakness intentionally justified rather than eliminated is an **accepted** finding. Acceptance does not lower the computed PAIN.
- A weakness reduced but not eliminated by a compensating control is **partially mitigated**: the grade and effect are re-evaluated against the *residual* condition on cited evidence, and both the pre- and post-mitigation N-ratings are recorded. This matches VER's expectation that a remediation clock is satisfied by partial mitigation to a lower N-rating.

The scoring algorithm is never weakened to compensate globally for scanner noise. Disposition changes the finding's status, not its original arithmetic. When partial mitigation changes the underlying condition, the residual PAIN is a second calculation against that changed condition; it does not rewrite the pre-mitigation score. Noise and exceptions are resolved one finding at a time against evidence, while the scoring method remains deterministic.

[5] FedRAMP, *VER* (*Evaluation Factors*): FedRAMP/2026-markdown@2a75592.
[6] FedRAMP, *VDR* (*Mitigation and Remediation Expectations*): FedRAMP/2026-markdown@2a75592.
[7] FedRAMP, *Definitions* (*Likely Exploitable Vulnerability*): FedRAMP/2026-markdown@2a75592.

4 The profile-derived asset band

IN PLAIN TERMS

The same weakness matters more on a crown-jewel system than on a scratch sandbox, so the first thing the method needs is the affected asset's security-impact profile. It reuses the CVE method's answer exactly: each asset carries separate requirements for secrecy, correctness, and uptime. This addendum projects those three values into one High, Medium, or Low band for its benchmark-finding matrix. The projection is one-way: it never replaces the underlying vector. If an asset has never been classified, it is treated as High-band until someone proves otherwise.

The affected asset's Security Requirements (CR, IR, AR) are the CVSS Environmental multipliers used throughout the CVE memo: Low = 0.5, Medium = 1.0, High = 1.5. The method summarizes them into a single band from their mean:

$$\text{asset\_mean} = \tfrac{1}{3}(\text{CR} + \text{IR} + \text{AR}), \qquad \text{band} = \begin{cases} \text{Low} & \text{asset\_mean} < 0.75 \\ \text{Medium} & 0.75 \le \text{asset\_mean} < 1.25 \\ \text{High} & \text{asset\_mean} \ge 1.25 \end{cases}$$

The attainable means are sixths (0.5, 0.667, ..., 1.5), so no threshold is ever hit exactly. The formula is the derivation; the rule an operator actually needs to remember is plainer:

DEFINITION

The band in plain language. An asset is **High** when at least two of its three requirements are High and none is Low; **Low** when at least two are Low and none is High; **Medium** in every other case. A single Low requirement demotes two Highs to Medium — `HHL` is Medium, not High.

Mixed vectors like `HHL` are valid and must remain representable even when they are uncommon in a particular estate. This addendum uses the band only because benchmark findings ordinarily lack the C/I/A technical-impact vector needed for objective-by-objective alignment. A conforming implementation MUST retain the independent CR/IR/AR profile, MUST NOT reconstruct or overwrite that profile from the band, and SHOULD expose the projection in its audit record. Direct per-objective assignment remains fully supported upstream; the band is a lossy convenience for this matrix, not an asset-classification system.

Missing or unresolved asset metadata resolves to `HHH` → High. Asset classification is provider-controlled, so its absence fails safe and loud: an unclassified asset scores at the top of the band and surfaces itself for classification rather than hiding. This is the same fail-safe doctrine the CVE memo applies; the addendum inherits it unchanged.

Affected-resource substitution. A benchmark finding is often detected at one resource but really endangers a higher-band one. A finding that reasonably affects a higher-band dependent resource, an administrative plane, a shared credential store, backup or logging infrastructure with reach into higher-band assets, or a control plane is evaluated *against that higher-band resource*, not against the point of detection. Substitution is governed by that named trigger list so it is not an open-ended judgment. It carries a firm audit obligation: every substitution records the substituted resource and the rationale, and the decision is recorded *both* when it is applied *and* when it is declined (Section 9). A full asset-adjacency graph would make substitution table-driven; it is noted as an optional refinement, not a baseline input, because the baseline may not assume metadata ordinary inventory does not provide.

IN PLAIN TERMS

Substitution in one example. A STIG finding lands on a bastion host. The bastion stores no customer data and is classified L/L/L — Low on every dimension — so on its own nameplate it would band Low. But the bastion is the sole administrative path to the production database, which is classified H/H/H. The finding is scored against the *database's* High band, not the bastion's Low band, because what the weakness actually endangers is the database behind it. The audit record captures the substituted resource (the database) and why (the bastion is its only admin path). And if a reviewer looks at the same finding and *declines* to substitute — judging the weakness genuinely confined to the bastion — that decision is recorded too, with its basis, so an assessor can challenge either call rather than only the ones that moved the score.

5 Finding grade and provenance

IN PLAIN TERMS

Next the method needs to know how serious the failed check is on its own terms. Where a trustworthy severity exists — a DISA STIG category, or a scanner whose severity scheme is documented and stable — it is used. Where none exists, as with raw CIS Pass/Fail, the finding takes a middling "Moderate" by default. That default is a deliberate, disclosed calibration, not a shrug: it reflects that most benchmark checks really are middling, and that stamping hundreds of hardening gaps as top-severity would bury the few that are genuinely dangerous.

The method assigns each finding one of three grades:

Grade	Sources
Major	STIG CAT I; governed scanner Critical/High
Moderate	STIG CAT II; governed scanner Medium; **unscored `Fail`** (the calibrated prior)
Minor	STIG CAT III; governed scanner Low

Table 1: Finding grade by governed severity source.

Provenance is resolved through a pinned, versioned **source registry**, and this is what keeps the grade from being provider-gameable. The registry records, per benchmark source, whether it is *structurally unscored* (CIS — eligible for the Moderate prior) or carries a *governed mapping* (with the scanner's full severity vocabulary at a pinned version). A scanner on the published governed list is governed by default and cannot be un-elected. DISA STIG categories are authoritative. The starter governed list is DISA STIG CATs, Google Security Command Center category severities, and AWS Foundational Security Best Practices control severities.

Two rules make the direction of every default explicit:

- A blank or unrecognized severity *from a scored, governed source* fails loud to **Major**. A provider cannot suppress a governed High and quietly ride the Moderate prior — stripping a governed severity fails *up*, not down.
- The Moderate prior applies *only* to registry-marked structurally-unscored sources. A provider can neither self-escalate nor self-reduce a governed severity, and CIS Level 1/2 profiles are never read as severities.

Why the prior is Moderate, in full. The default deserves its complete justification on the record, because a worst-case default would seem safer and is not. Missing benchmark severity is *structural*, not a provider omission: CIS publishes no severities at all, so treating their absence as a fail-safe worst case would be miscalibration rather than caution — it punishes a property of the benchmark, not a gap in the provider's evidence. This is the sharp line between the two kinds of unknown the method handles: a *provider-controlled* unknown (an unclassified asset, an unknown firewall state, an unknown agency scope) fails safe and loud, but a *structurally absent* input (benchmark severity) takes the calibrated prior. Beyond that principle, two empirical facts support Moderate specifically. CAT II is the mode of real benchmark content — most checks are genuinely middle-weight hardening items. And labeling hundreds of ordinary hardening deviations "potentially debilitating multi-agency events" would destroy the incident signal for the findings that actually deserve it. A default that cries wolf on everything protects nothing.

The escalation override. The prior can be wrong in the minority, and the method gives that minority a governed home rather than reopening the flood. Documented evidence that a specific finding yields direct High C/I/A impact on the (possibly substituted) resource promotes its grade one step (Moderate → Major). The step is audited, fenced by the same registry and assessor-review governance as the source registry, and expected to be rare. An encryption-at-rest CAT II on a shared multi-agency datastore is the canonical case: genuinely under-claimed at its default grade, and now escalable on evidence.

6 Customer effect and PAIN

IN PLAIN TERMS

Now the two halves meet. The grade (how serious the check is) and the profile-derived asset band (the aggregate strength of the asset's three requirements) combine in a small lookup table to produce a plain-English effect word. That word, together with whether the system serves one agency or several, selects the N-level. The band is a ceiling: a serious check on a low-requirement box cannot climb higher than the asset can support. And one boundary is drawn on purpose — the line between an "incident" and "maintenance" sits exactly where it does to keep the routine flood out of the incident lane.

The grade and the asset band select a customer-effect word:

Grade	Asset High	Asset Medium	Asset Low
Major	Debilitating	Disruptive	Narrow
Moderate	Narrow	Narrow	Minimal
Minor	Minimal	Minimal	Minimal

Table 2: Customer-effect matrix (grade × asset band).

The effect word and the asset's agency scope select the PAIN level, using FedRAMP's fixed effect-and-scope semantics:

Customer effect	Single agency	Multi-agency
Debilitating	N4	N5
Disruptive	N3	N4
Narrow	N2	N2
Minimal	N1	N1

Table 3: Effect × agency scope → PAIN. N5 is reachable only for a Debilitating multi-agency finding.

Unknown agency scope resolves to *single*-agency. This is the one deliberate exception to the method's loud-default doctrine, made for signal-to-noise: benchmark findings arrive in the hundreds, and a multi-agency default on every untagged asset would move whole scan populations up the incident column on missing metadata alone. The exposure is bounded and disclosed. On a High-band asset the default costs one level — N4 rather than N5, both incident-class in Class D, so prioritization survives intact — and on a Medium-band asset a Major finding lands N3 rather than N4, the one place the default crosses the incident line. The offset is operational rather than scoring-side: scope is provider-controlled metadata that tag-enforcement policy can require at provision time, and scanners surface untagged resources in their own reporting, so the unknown-scope population is visible and reducible. The audit record carries the "defaulted" marker either way. The band acts as a ceiling in Table 2: severity never raises effect above what the asset can support, so a Major finding on a Low-band asset is still only Narrow. And the Moderate → Minor grade transition never drops the effect by more than one band (exactly one, unless the effect is already Minimal), which keeps the matrix monotone.

Two observations the tables make plain. First, **the multi-agency column has no N3.** This is FedRAMP's own definition, not an artifact of this method: N3 is defined as an effect disruptive to a single agency, and disruption of more than one agency is N4 by definition. No classifier could put a genuinely multi-agency Disruptive effect on N3 and still be faithful to the rule. The line between an incident (N4 and above) and maintenance (N2 and below) therefore sits exactly on the Disruptive boundary, and the effect matrix is calibrated with that boundary in view.

That said, N3 is not unreachable on shared infrastructure — because the scope input is the agency scope of the finding's *effect*, not the asset's nameplate tenancy. A finding on a shared component whose exploitation is demonstrably contained to a single agency's tenancy — a per-tenant bucket-prefix policy error, a tenant-scoped IAM binding that leaks only within one tenant — scores single-agency, so a Disruptive such finding lands on N3, not N4. That single-agency assertion on a shared asset is exactly the case the audit record already requires a tenancy basis for (Section 9); the containment must be evidenced, not asserted. One boundary matters here: the single-agency default applies to *unknown* scope — an untagged asset — not to assets known to be shared. A finding on a shared control plane whose blast radius is not demonstrably contained still resolves to multi-agency; missing containment evidence on a known-shared asset is not an unknown scope.

Second, **the Moderate/High cell is a disclosed calibration knob.** It reads Narrow, not Disruptive. The alternative — Disruptive — would put every CAT II and every raw CIS `Fail` on a High-band multi-agency asset at N4, which is incident-class in Class D. That is precisely the flood this design exists to prevent, so the knob is set to Narrow deliberately and disclosed here rather

than buried. The choice has a named residual risk, and the method owns it rather than papering over it:

DEFINITION

Named residual risk (the N5↔N2 cliff on High assets). Because Moderate/High caps at Narrow → N2, a genuinely dangerous CAT II on a crown-jewel multi-agency asset — the standing example is an account-lockout policy gap on an internet-exercisable admin plane, brute-forceable in practice — is capped at N2 unless the escalation override (Section 5) is invoked on evidence. The gap between that N2 and the N5 the finding might deserve is the calibration's known cliff. It is not closed by moving the matrix cell (that would let exercisability set PAIN, the double-count the method forbids); it is closed, when it applies, by the evidence-gated grade escalation, and it is owned by the certifying provider.

7 Internet exercisability: the remediation column

IN PLAIN TERMS

The N-level says how bad; this step says how fast. FedRAMP tightens the clock for findings that are likely exploitable and internet-reachable. For benchmark findings we cannot measure those directly, so the method uses one honest, conservative proxy: can an unauthenticated stranger on the internet actually exercise this failed setting, as it stands, without first getting something the finding itself does not hand them? If yes, it rides the fast clock. This test is a practical stand-in, not a new definition of FedRAMP's reachability — and it only ever errs toward calling something *not* exercisable, never the other way.

The normative test is a single sentence: **a benchmark finding is internet-exercisable if and only if an unauthenticated internet actor can exercise the failed condition as it currently stands, with no precondition the finding does not itself supply.** It follows the same separation of the CVE memo's exploitability and reachability axes.

DEFINITION

Internet-exercisability is an **operational test for benchmark findings**. It is *not* a redefinition of FedRAMP's Internet-Reachable Vulnerability status, and it must not be read as one. The relationship is one-directional and stated as a soundness claim: **exercisable $\implies$ IRV, never the converse.** A finding the test calls exercisable is genuinely internet-reachable; a finding it calls not exercisable may still be IRV by an indirect path the proxy cannot see. The baseline is therefore *sound but incomplete* with respect to IRV — it can only ever *under*-select the fast clock, never over-select it — and the evidence-backed overrides below exist to catch the cases it misses.

One test sets both letters (LEV = IRV). For benchmark findings this single determination is the whole column story. A failed configuration needs no separate exploit-likelihood analysis, because the misconfiguration *is* the exploitable state — there is no exploit to develop, no payload to craft; the weakness is standing open the moment the scanner observes it. Exercisable therefore means likely exploitable *and* internet-reachable at once, and the finding rides LEV+IRV. Not exercisable means NLEV. LEV without IRV — the internal-actor column — arises only through the evidence-backed overrides at the end of this section, never from the baseline.

DEFINITION

Alternative exploitability sources. LEV = IRV is deliberately positioned as the adoptable default, not the only lawful answer. There is no EPSS for misconfigurations, and no individual provider should

have to build a bespoke exploitability-evaluation framework for compliance findings — that duty does not scale, and a hundred unvalidated bespoke frameworks would each be a soft lever on the clock. A richer evaluation is nevertheless permitted, from either direction: an EPSS-equivalent developed for compliance findings, or an equivalent metric native to the scanner in use. Either is admissible under the same discipline the method already applies to severity sources: it must be evidence-backed, and its logic for aligning the metric to the LEV/NLEV columns must be documented, versioned, and disclosed for assessor review. One qualification is doing real work in that sentence: the metric must actually measure *exploit likelihood*. A score that is a function of internet reachability and the profile-derived asset band re-derives the two inputs this method already computes deterministically, and adopting one substitutes an opaque restatement for the determination table without adding the missing ingredient. The metric value emitted at evaluation time is captured in the audit record as an input, so the column remains reproducible from the record even when the metric itself is recomputed upstream. Absent such a documented alignment, the baseline default governs.

Per-family operational rules. The test is evaluated by benchmark family, each family using only data ordinary scanners and inventory provide:

- **Host benchmarks, OS-level** (RHEL/Windows STIG and CIS): exercisable iff the administrative/login plane is world-open, tested against a governed port set (default $\{22, 3389, 5985, 5986, 5900, 10250\}$). The `admin_plane_open` bit is not scanner-emitted metadata — it is the output of a *required join* against governed firewall/load-balancer inventory, the same feed the CSP benchmark's own network controls read. There is no per-rule mapping. Unknown firewall state resolves to open: firewall state is a discoverable unknown, and the doctrine is to score discoverable unknowns loud to force the inventory join. A WAF/ALB-fronted surface counts as open — a WAF is mitigation, not closure.
- **Host benchmarks, application-level** (web/db STIGs): exercisable iff that application's own service surface is internet-open, by the same join.
- **Cloud-configuration benchmarks:** exercisable iff the finding category is *exposure-class* under the governed determination table below.

IN PLAIN TERMS

Answering the stranger-on-the-internet question for hundreds of cloud finding types, without a human judging each finding one at a time, is the job of a lookup sheet. For each benchmark there is one pre-approved list of every finding type the scanner can emit, each stamped with one of three labels. *Exposed*: the finding itself means something is open to the internet — a public bucket — so the stranger can poke it and the fast clock runs. *Depends*: the finding is only dangerous if the thing it is attached to is public — weak SSL on a database matters only if the database has a public IP, which the same scan already tells us — and when we cannot tell, we assume public. *Internal plumbing*: passwords, logging, admin hygiene — a stranger on the internet cannot touch it, the normal clock runs, and most findings land here. Scoring is then just looking the finding type up on the sheet; everything below is about how the sheet is built, and how it is kept honest.

The determination table. For cloud benchmarks the normative object is a governed, content-hashed table published per (benchmark, version, scanner). The table assigns every category identifier in the version's vocabulary an effective classification: *exposure-class* (always exercisable), *surface* (exercisable iff the attached resource is public, joined from the same scan's exposure findings; unknown attachment resolves to public), or *identity/operations-plane* (not exercisable). The

classification reads the category identifier the scanner emits — never the rule prose, and never per-resource IAM analysis — so the determination is a lookup, and a second implementer reproduces any result from exactly two inputs in the audit record: the table hash and the category. A scanned category *absent* from the table's vocabulary — a scanner's monthly new addition, say — fails loud to LEV+IRV and is flagged unreviewed.

That is the whole normative requirement: the table, its three classes, the fail-loud rule, and the governance below. *How a canonical table is built* — what makes a category exposure-class in the first place — is implementation guidance:

IMPLEMENTATION GUIDANCE (NON-NORMATIVE)

The recommended generator is token and literal-pattern matching on the identifier. Normalize — uppercase, split on non-alphanumeric runs — and classify exposure-class if any token is one of {`PUBLIC`, `OPEN`, `ANONYMOUS`, `INTERNET`, `WORLD`}. Token boundaries make the match safe across scanner grammars: `OPEN_SSH_PORT` matches on the `OPEN` token while `OPENSSH_CONFIG` does not, and an AWS Config identifier such as `s3-bucket-public-read-prohibited` normalizes to the same tokens as its GCP cousin. Before tokenizing, match two literal patterns: an identifier containing `0.0.0.0` or `::/0` is exposure-class — caught by literal substring precisely because normalization would split `0.0.0.0` into a bare `0` and lose the meaning. So `sg-allow-ingress-from-0.0.0.0-0` matches, while the plain numeric segments of `TLS_1_0_ENABLED` do not. Because the generator reads scanner-emitted identifiers, a provider cannot game it by renaming; because its output is frozen into the reviewed, hashed table, the generator itself never runs at scoring time. In practice a canonical table is serialized as review deltas against the generator — `add`, `remove`, and `surface` entries plus the `vocabulary` — so review effort concentrates on deviations. A category left untagged in that serialization is not a gap: it is one the review affirmatively classified identity/operations-plane.

Two tokens that look tempting — `EXTERNAL` and `UNRESTRICTED` — are deliberately *not* in the auto-match set. Their genuine true positives are already carried by `PUBLIC`/`OPEN`/`INTERNET` or by explicit `add` entries, while their false-positive rate is high (`SQL_EXTERNAL_SCRIPTS_ENABLED` is not an internet exposure; `API_KEY_APIS_UNRESTRICTED` scopes a key, not a surface). Keeping them out holds the auto-match vocabulary tight and moves the burden of proof onto a reviewed table entry. Screening a benchmark's full category list for token collisions when the table is built is a sound practice, not a normative requirement. So is richer analysis: automated or agentic analysis of a benchmark's category list to pre-classify exposure relevance for triage is permitted and useful, but it is not required — token and literal-pattern matching against a reviewed table meets the bar.

Governance. The governance around the table is what makes it trustworthy rather than a hidden lever:

- Canonical tables are published per (benchmark, version, scanner) and **content-hashed**; the hash travels in the audit record. A version *string* enforces nothing; a hash does.
- Any provider deviation from the canonical table is enumerated per-entry with a rationale and flagged for assessor review.
- Reclassifying a category out of exposure-class is a hard assessor-review trigger — it is the one move that silently slows a clock, so it never happens silently.
- Keeping the tables current is an operating requirement with a monthly new-category diff, not a once-per-version event.

The identity/operations-plane class deserves its justification stated once, because it is where most of a benchmark lands. The test's own precondition clause supplies it: a user-managed service-account key requires possessing the secret, and the public cloud control plane alone gives an internet actor nothing to exercise. A data-plane grant to `allUsers` or `allAuthenticatedUsers`, by contrast, *is* exposure-class regardless of the category name — `allAuthenticatedUsers` is satisfied by any

self-registered account, which the unauthenticated test treats as self-suppliable.

Overrides. The overrides carry the cases the sound-but-incomplete baseline misses, and each is evidence-backed, audited, and expected to be uncommon:

- **LEV+NIRV** when evidence shows a likely tenant, internal, adjacent, or local actor can exercise the condition. The canonical case is the MFA-not-enforced family (Section 10).
- **LEV+IRV** when evidence shows public usability (for example a known-leaked key, at which point incident response applies), or on transitive-reachability evidence — an indirect internet payload path, mirroring the CVE memo's transitive branch — which gives the VER indirect-payload case a home in the IRV direction.

The column then selects itself, completing the LEV = IRV rule stated at the top of this section: LEV+IRV if internet-exercisable or so overridden; LEV+NIRV on an evidence-backed internal-exploitability override; NLEV otherwise. The determination is stated once more for emphasis: **exercisability changes the column only; it never changes PAIN.** An affirmative not-exercisable determination (any `False` on `admin_plane_open`, `service_open`, or `attached_resource_public`) requires an enforced-state evidence pointer — a security-group or firewall rule identifier, a listener binding — because the absence of an observation is not the same as enforced closure.

8 Deadlines

IN PLAIN TERMS

The last step is a table lookup. The PAIN level, the remediation column, and the provider's assurance Class pick a single cell, and that cell is the number of days allowed. The numbers are FedRAMP's, not ours — the very same matrix the CVE method uses — so nothing benchmark-specific is invented here.

The deadline is deadline = M[Class][PAIN][column], drawn from the pinned `VDR-TFR-PVR` matrix reproduced in the CVE memo's appendix on remediation deadline matrices (days; 0.5 = 12 hours; Classes A and B share a schedule; N1 carries no FedRAMP deadline). The full tables are not restated here; only the single N2 row the worked examples use is reproduced for convenience:

	Class A / B			**Class C**			**Class D**		
PAIN	L+I	L+N	NLEV	L+I	L+N	NLEV	L+I	L+N	NLEV
N2	96	160	192	48	128	192	24	96	192

Table 4: The single `VDR-TFR-PVR` row used by the worked examples, reproduced from the CVE memo's appendix. The complete N5–N2 matrix lives there and is not duplicated.

The anchor case ties the whole chain together. A CAT II on a Medium-band, internet-exercisable host is Moderate × Medium → Narrow → N2, LEV+IRV, which is **24 days in Class D** (48 in Class C, 96 in Classes A/B) — and 192 days if it were not exercisable. Exercisability moved the clock by an order of magnitude and left the N-level untouched.

Where these clocks are actually won. A remediation clock is the wrong place to fight compliance-finding volume, and the deadline machinery should not be mistaken for the strategy. Compliance findings overwhelmingly open at build and deploy time, not part-way through operations and maintenance: a bucket is created public, a firewall is opened, a role is over-granted.

The practical lever for N4/N5 volume is therefore *pre-deployment* — infrastructure-as-code and policy analysis that evaluates the same two inputs this method needs (asset security-impact profile × finding category) before the resource exists. A public bucket is an acceptable pattern in some designs; a public bucket with a High-band data profile is precisely the thing to block at deploy time rather than remediate on a 12-hour clock afterward. The timeframe matrix runs from completed evaluation and governs what escapes into production; prevention is what keeps the incident-class population near zero in the first place. The back-test that follows should be read in that light — a small fast-clock count is the goal, and shifting the check left is how it stays small.

9 The minimum audit record

IN PLAIN TERMS

Everything the method decides has to be reconstructable later from what it wrote down. Two reviewers handed the same finding, the same pinned tables, and the same evidence must reach the same answer — so the audit record captures not just the result but every input and every judgment call, including the ones where the method chose *not* to act.

Every output is reproducible from a single record. It captures: the benchmark and version; the rule or category identifier and the scanner result; the affected asset, its uncapped security-impact profile, and the profile's derivation method, source, and rationale; the affected-resource substitution and rationale when applied, *and* the substitution decision and basis when declined; the CR/IR/AR and derived band; the agency scope, with a "defaulted" marker if it was unknown; a tenancy basis (agencies served, plus evidence) for any single-agency assertion on a shared-service asset; the severity and its provenance (the grade and why the source is governed or unscored); the customer effect and PAIN; the exercisability determination together with the determination-table content hash and the enumerated provider delta; an enforced-state evidence pointer for any affirmative not-exercisable determination; the column and any override evidence reference — plus, when an alternative exploitability source is used, the metric value emitted at evaluation time and the version of its documented column alignment; and the Certification Class, the evaluation completion time, and the resulting deadline. It also records the current disposition and its evidence when analysis has been completed. The original PAIN remains in the record for every scored finding; when a residual condition is re-scored after partial mitigation, both the pre- and post-mitigation PAIN are retained.

10 Worked examples

IN PLAIN TERMS

The method run by hand on the cases that decide whether it is calibrated right — including the two it deliberately handles differently from what a first glance suggests (the service-account key, and MFA).

Example 1 — user-managed service-account key (`USER_MANAGED_SERVICE_ACCOUNT_KEY`). Present in the vocabulary with no exposure or surface classification: identity/operations-plane → **NLEV**. The public GCP API plane is not the finding's exposure; an internet actor lacks the secret, so the unauthenticated test fails at its precondition. Overrides remain available: NIRV on evidence of broad internal availability plus harmful permissions, IRV only on evidence of public disclosure.

Contrast — API keys (`API_KEY_APPS_UNRESTRICTED`). The same possess-the-secret reasoning does *not* transfer. Client-embedded GCP API keys (Maps, Firebase web and mobile) are public by design; the finding is precisely that the compensating restriction on an already-public key is missing. These categories are therefore `surface`, not `remove`: the attachment is "key is client-distributed," unknown distribution defaults to exercisable, and a down-override to NLEV requires evidence the key is server-side-only and network-scoped. Two findings that both contain `UNRESTRICTED` split on their actual exposure, not on a shared word.

Example 2 — public bucket (`PUBLIC_BUCKET_ACL`). Exposure-class in the determination table (matched on the `PUBLIC` token by the generator) → **LEV+IRV**. Governed SCC severity High → Major; on a High-band multi-agency data asset the effect is Debilitating → **N5** → **12 hours in Class D**. That is an incident, and the method calls it one.

Example 3 — `SSL_NOT_ENFORCED` on a Cloud SQL instance where `SQL_PUBLIC_IP` is Compliant. A `surface` category: exercisable only if the attached resource is public. The same scan shows the instance has no public IP, so the attachment joins to False → **NLEV**. This is the permitted category-level inference — when the attaching exposure control is fully Compliant in the same scan, its resources are non-public by evidence, not assumption.

Example 4 — CAT II on a public web VM with admin ports closed. An OS-level host finding. The VM serves the public over 443, but the administrative plane is closed by the firewall join, so the finding is not exercisable via 443 → **NLEV**. On a Medium-band asset it is Moderate × Medium → Narrow → N2 → **192 days in Class D**. The application being internet-facing does not by itself make an OS-hardening deviation internet-exercisable.

Example 5 — MFA not enforced. The strict test says not exercisable: an internet actor still needs a valid credential, which the finding does not supply. This is the canonical override case named in Section 7. A provider facing a likely tenant or internal actor SHOULD consider the LEV+NIRV override (or a deliberate LEV+IRV where credential stuffing against a public sign-in is the realistic threat), recording the evidence once as a standing family rule rather than finding-by-finding.

11 Back-test results

IN PLAIN TERMS

The real test of a calibration is whether it floods the incident queue on a genuine scan. It does not. Run against a real, sanitized CIS Google Cloud scan of nearly fourteen hundred findings — and under the harshest asset assumption, treating every asset as High-band — only fourteen findings land on a fast clock, and the incident lane stays essentially empty. The bulk sits at the low, slow end, which is exactly where routine hardening gaps belong.

The method was back-tested against a sanitized real CIS Google Cloud Platform Foundation 2.0 scan of **1,378 findings** (Class D, multi-agency, from Google Security Command Center). To stress the calibration, the run assumes the *High* asset band uniformly — the loudest assumption available. Even so, the distribution is dominated by the slow, low end:

PAIN	Column	Findings	Class-D deadline
N5	LEV+IRV	5	12 hours
N5	NLEV	2	8 days
N2	LEV+IRV	9	24 days
N2	NLEV	736	192 days
N1	NLEV	626	no deadline
Total		**1,378**	

Table 5: Back-test distribution on the sanitized CIS GCP 2.0 scan (Class D, multi-agency, uniform High-band assumption). Only exposure-class findings ride the LEV+IRV fast clock.

Two properties matter. First, there is **no N4/N5 flood**: only seven findings reach incident-class at all, and just five ride the 12-hour clock, on the loudest possible asset assumption. Second, the fast clock is scarce and earned: $5 + 9 = 14$ findings ride LEV+IRV, exactly **1.0%** of the population, and every one of them is an exposure-class category. The remainder — the 736 N2 and 626 N1 findings — are the ordinary hardening backlog, correctly parked on the slow clocks where they belong. Dropping the asset assumption to Medium shifts the incident-class findings from N5 to N4 (a slightly slower incident clock) and leaves the N2/N1 bulk untouched, which is the intended behavior: the asset band moves the row, never the column.

12 Known limitations

IN PLAIN TERMS

The method is an honest proxy, and honest proxies have known blind spots. It can shout a little too loudly about harmless findings that merely sit on an exposed host, and it can under-rate a finding an insider could exploit unless someone supplies the evidence to override it. It is a calibrated triage tool for benchmark findings, not a substitute for a full exploitability analysis, and it is never presented as one.

The method's approximations are disclosed, not hidden:

- **The profile-derived band is intentionally lossy.** Benchmark findings ordinarily do not carry a C/I/A technical-impact vector, so this addendum cannot align a failed check independently to CR, IR, and AR as the CVE method does. The band preserves a deterministic aggregate for the lookup matrix, but mixed profiles can be represented imperfectly. The full profile remains authoritative and the band must never overwrite it; a future finding format with dimensional impact metadata could remove this approximation.
- **It over-prioritizes some non-exercisable findings on exposed surfaces.** The host proxy rides the admin-plane bit for the whole host, so a bastion's entire CAT II set can ride LEV+IRV even though most of those findings are not individually internet-exercisable. This over-generation is the named, disclosed error shape of the host proxy, and it is the accepted price of refusing per-rule mappings. It is also less wrong than it first looks: on a host whose login plane faces the internet, hardening gaps compound on the same attack surface, so treating the set as urgent carries real signal. And the flood collapses through a single remediation — close the admin plane or gate it behind a VPN or identity-aware proxy, and every finding on that host returns to NLEV at the next scan. The over-generation points at exactly the fix that removes it.

- **It under-prioritizes internally exploitable findings absent an override.** Because the baseline exercisability test is sound but incomplete (exercisable $\implies$ IRV, never the converse), a finding an insider, tenant, or adjacent actor could exploit stays on the slow clock until the evidence-backed override is applied. The MFA family and tenant-isolation families are the standing examples, handled as family rules rather than per-finding.
- **The unclassified-asset and unknown-firewall defaults are loud on purpose, and that is itself a flood vector.** Operating without an inventory classification feed or a firewall feed will over-score, by design — the loud default exists to force those joins. Inventory classification is an operating precondition, not a silent assumption.
- **It is not a full LEV evaluation.** The method is a calibrated, deterministic triage for benchmark findings from the data scanners actually emit. It is never described as equivalent to a full likely-exploitability analysis, and where a finding warrants that depth, the override and disposition paths route it there.

IN PLAIN TERMS

What to take away. The method scores benchmark findings using only what scanners and ordinary inventory already emit — a rule identifier, a Pass/Fail, an asset tag, a firewall feed. Every unknown the *provider* controls — an unclassified asset, an unknown firewall state — scores loud, so silence never makes a problem look smaller than it is. The method quietly picks a default in exactly two places, both calibrated for signal-to-noise and both disclosed here in full rather than hidden: the severity of a benchmark that structurally publishes none, where an unscored `Fail` becomes Moderate, and an unknown agency scope, which defaults single-agency and leans on tag enforcement to keep the untagged population shrinking. Those two choices are the reason a real scan of 1,378 findings produces fourteen fast-clock items instead of a wall of false alarms.

A Worked determination table: CIS GCP Foundation 2.0

The determination table below is the canonical artifact for the CIS Google Cloud Platform Foundation 2.0 benchmark as emitted by Google Security Command Center. It is what the back-test of Section 11 ran against. In production the table is content-hashed and the hash is carried in each finding's audit record; the vocabulary is the benchmark version's complete category list, abbreviated here to the entries the classification logic turns on.

The recommended generator classifies the pure-exposure categories automatically — their identifiers carry an exposure token — and review confirmed each, so they carry no delta entry and are always exercisable:

Category	Matching token	Result
`PUBLIC_BUCKET_ACL`	`PUBLIC`	LEV+IRV
`PUBLIC_DATASET`	`PUBLIC`	LEV+IRV
`PUBLIC_SQL_INSTANCE`	`PUBLIC`	LEV+IRV
`KMS_PUBLIC_KEY`	`PUBLIC`	LEV+IRV
`OPEN_SSH_PORT`	`OPEN`	LEV+IRV
`OPEN_RDP_PORT`	`OPEN`	LEV+IRV

Table 6: Exposure-class categories (generator-matched, confirmed on review).

The `surface` entries are exercisable only when the attached resource is public, joined from the same scan's exposure state; unknown attachment resolves to public (fail-safe toward the fast clock):

surface category	Attachment tested
SSL_NOT_ENFORCED	Cloud SQL instance public (via SQL_PUBLIC_IP)
WEAK_SSL_POLICY	fronting load balancer public
SQL_NO_ROOT_PASSWORD	Cloud SQL instance public
PUBLIC_IP_ADDRESS	firewall/authorized-network state (reachability, not an exercisable surface)
SQL_PUBLIC_IP	firewall/authorized-network state
API_KEY_APPS_UNRESTRICTED	key is client-distributed (public by design; unknown → exercisable)
API_KEY_APIS_UNRESTRICTED	key is client-distributed

Table 7: `surface` categories: exercisable iff the attached resource is public.

The `add` and `remove` sets are both empty for this version: the generator covers every pure-exposure category, and after the API-key categories were moved to `surface` there is no generator false positive left to strip. Every other category in the vocabulary — the logging, monitoring, IAM, CMEK, and SQL-hardening families such as `USER_MANAGED_SERVICE_ACCOUNT_KEY`, `MFA_NOT_ENFORCED`, `DNSSEC_DISABLED`, `IP_FORWARDING_ENABLED`, and `COMPUTE_SERIAL_PORTS_ENABLED` — is identity/operations-plane and resolves to NLEV, with the MFA and tenant-isolation families carrying the standing overrides of Section 7. Integrity-hardening categories whose exploitation needs an off-path or race precondition (`DNSSEC_DISABLED`, `RSASHA1_FOR_SIGNING`) fail the unauthenticated-exercise test and take the NLEV default with an override available. A category scanned but *absent* from the vocabulary fails loud to LEV+IRV and is flagged unreviewed.

Normative sources

[DEF] FedRAMP, *FedRAMP Definitions*. Definitions cited: *vulnerability* (FedRAMP/2026-markdown@2a75592), *Likely Exploitable Vulnerability* (FedRAMP/2026-markdown@2a75592). Commit of 2026-06-24; accessed 2026-07-16.

[VER] FedRAMP, *Vulnerability Evaluation and Reporting (VER)*. Sections cited: *Estimate Potential Agency Impact* (FedRAMP/2026-markdown@2a75592), *Evaluation Factors* (FedRAMP/2026-markdown@2a75592). Commit of 2026-06-24.

[VDR] FedRAMP, *Vulnerability Detection and Response (VDR)*. Section cited: *Mitigation and Remediation Expectations* (FedRAMP/2026-markdown@2a75592); and the `VDR-TFR-PVR` remediation matrix, reproduced in the CVE memo's appendix. Commit of 2026-06-24.

[CIS] CIS, *CIS Benchmarks FAQ* (Level 1 and Level 2 profiles). cisecurity.org/cis-benchmarks-faq; accessed 2026-07-16.

[CVEM] M. Venne, *A Deterministic, CVSS–Environmental Method for FedRAMP VDR/VER Vulnerability Prioritization*, stackArmor, July 2026. The core memo this addendum extends, supplying the asset model, the disposition overlay, and the `VDR-TFR-PVR` deadline matrix.

[BOD] CISA, *Binding Operational Directive 26-04*, 2026-06-10. The mandating authority for the VDR/VER timeframes.