

#	Category	Field / Element	JSONPath / JSON Pointer	Purpose / Description	Normalization / Validation Rules	Derived or Analytical Metrics
1	Core Metadata	fields: id, state, published, lastModified, providerMetadata	\$.cveMetadata.*, \$.containers.cna.providerMetadata	CVE identifiers, status, and CNA info.	Ensure consistent ID format (CVE-\d{4}-\d+); validate CNA attribution.	Record lineage; publication latency.
2	Description(s)	\$.containers.cna.descriptions[*].value (+ .lang)	Textual narrative of vulnerability, with language tags (BCP-47).	Enforce language codes; trim HTML tags; flag extreme length or non-ASCII noise.	char_len, token_len, html_tag_count, text anomaly scores.	
3	Affected Software / Platforms	\$.containers.cna.affected[*]	Lists impacted vendor, product, versions, platforms.	Require structured arrays; normalize vendor/product names; reject substring matches.	Ecosystem alignment via purl; product-level vulnerability graph.	
4	Source / CNA / Credits	\$.containers.cna.providerMetadata, \$.containers.cna.credits[*]	CVE issuer and contributors.	Validate CNA names against registry; ensure timestamps and contact fields exist.	Contributor diversity metrics; provenance reliability.	
5	Severity & Metrics (CVSS)	\$.containers.cna.metrics[*], \$.containers.adp[*].metrics[*]	CVSS scoring objects (v2/v3/v3.1/v4).	Check prefix-version match (CVSS:3.1/ ⓘ object); reject invalid tokens; Δ≥1.5 between sources → flag.	base_score, base_severity, temporal_score, env_score, Exploitability Index, Cross-Vector Drift.	
6	Vector String	fields: ...vectorString	Encoded attack/impact vector.	Regex validate full CVSS structure; order and enums must match version spec.	Parsed metric factors (AV, AC, PR, UI, S, C, I, A).	
7	Temporal / Environmental Scores	fields: ...temporalScore, ...environmentalScore	Temporal and environmental context modifiers.	Verify enums (E, RL, RC); check consistency with base vector.	Adjusted severity differentials; confidence weighting.	
8	CWE / Problem Type	\$.containers.cna.problemTypes[*].descriptions[*].cweId, .description	CWE identifier and weakness description.	Match ^CWE-\d+; allow "CWE-Other" / "NoInfo" with expiry; compare text similarity ≥ 0.85 to official CWE.	CWE Depth, Specificity Index, CWE Completeness Score.	
9	References	\$.containers.cna.references[*]	URLs for advisories, patches, etc.	Validate URL format and type; deduplicate.	Reference type coverage; vendor vs. third-party ratio.	
10	Languages	\$.containers.cna.descriptions[*].lang	Declared description languages.	Must be valid BCP-47 code.	Multilingual coverage stats.	
11	Text / Semantic Outliers	—	Description or metric fields.	IQR thresholds for char_len, non-ASCII%, or HTML tags.	Text anomaly score; linguistic quality metrics.	
12	Scoring Outliers	—	CVSS scores across CNAs.	Δ ≥ 1.5 between base scores across sources → anomaly.	Cross-source variance index.	
13	Identity Anomalies	—	Product/package names.	Require ecosystem-aware purl; no substring matching.	Product identity consistency rate.	
14	CWE-CVSS Fusion	CWE categories × CVSS vectors	Derived model.	Map CWE family to exploitability features; compute contextual multipliers.	Risk aggregation index; predictive inference model inputs.	
15	CVE-Core Tables	fields: cve_core, description, affected, metric, problemtype, ref	Canonical relational design.	Ensure referential integrity; unique id.	Enables cross-CNA joins and longitudinal analysis.	
16	Upstream Quality Feedback	—	—	Identify systematic schema or data issues.	Enforce HTML/diff filters; metric provenance; SLA for "Other/NoInfo".	Upstream feedback metrics; CNA quality ranking.
17	Validation Regex	patterns: <[>]+>, `(?m)^(?:diff --git	index [0-9a-f]{7,}	@@[^@]+@@"; CVSS v3.1 coarse pattern	Regex-based validators.	Match/strip patterns before ingestion.
18	Package Disambiguation Rules	—	Affected product fields.	Tokenize by [-_.]; lowercase; drop non-semantic suffixes; restrict by ecosystem.	Package-match precision; false-positive rate.	
19	Statistical / Outlier Framework	—	Entire dataset (/cves/YYYY/**)	Quantitative quality analysis.	Compute P50/P75/P90 annually; flag ≥ Q3 + 1.5 × IQR.	Annual anomaly index; data-quality heatmap.
20	Upstream Data-Quality Recommendations	—	—	Guidelines for CNAs / MITRE improvements.	HTML filter; provenance fields; SLA for CWE refinement; purl inclusion.	Reduction in downstream remediation workload.
21	Extraction Selectors	As defined in section 8	Deterministic selectors for JSON fields.	Maintain reproducible extraction paths and validation rules.	Pipeline reproducibility score.	