



PHOENIX
SECURITY

↑ 4.5x
growth

CVEs ASSIGNED: 0

106 campaigns · 1,868 IOCs

2024 / 2025 / 2026 H1



THE ACCELERATION: SUPPLY CHAIN ATTACK REPORT 2026

JULY REFRESH

ONE TOKEN. HUNDREDS OF PACKAGES. ZERO CVEs.

phoenix.security · Malware Package Intelligence 2026

PHOENIX MPI CORPUS · 106 CAMPAIGNS · 1,868 IOCS · 2018–15 JUL 2026 · JULY REFRESH

The **Acceleration**: How Supply Chain Attacks Went Industrial Across npm, PyPI, VS Code, and AI Agent Tooling

The Phoenix Malware Package Intelligence corpus now spans 106 tracked campaigns and 1,868 curated malicious-package IOCs, with a full-corpus population of 473,936 records. In the six weeks since the June baseline, the corpus roughly doubled. A new worm lineage (Miasma) established itself with a 72-hour adaptation cycle. Attackers began shipping validly-signed malware. Zero CVEs assigned during active exploitation, still.

Author: Francesco Cipollone, Phoenix Security**Originally published:** 7 June 2026**Updated:** 16 July 2026**Classification:** © 2026 Phoenix Security. All rights reserved.**106**

CAMPAIGNS TRACKED (2018–15 JUL 2026)

4.8×

IOC GROWTH, 2026 (TO 15 JUL) VS ALL OF 2025

56 / 106

CAMPAIGNS TOUCHING AI / LLM / AGENT TOOLING

15.6%

AI-AGENT SKILLS FLAGGED RISKY (VS 6.9% IDE EXTENSIONS)

1,868

INDEXED MALICIOUS PACKAGE IOCS (CURATED)

17×

2025 VS 2024, FULL-CORPUS DATED INCIDENTS

0

CVEs ASSIGNED DURING ACTIVE EXPLOITATION

26%

DEEP-SCANNED SKILLS RATED CRITICAL-RISK

What changed since the June baseline

The June baseline (published 7 June, cutoff 7 June) is preserved unedited so its numbers stay citable. This refresh queries the live `compromise_campaigns`, `campaign_package_links`, and `compromised_package_intel` tables again as of 15 July, adds six weeks of new intelligence, and audits the delta:

DELTA TABLE — JUNE 2026 BASELINE → JULY 2026 REFRESH

METRIC	JUNE BASELINE (CUTOFF 7 JUN 2026)	JULY REFRESH (15 JUL 2026)	DELTA
Tracked campaigns	59	106	+47 (corpus roughly doubled)
Curated indexed IOCs	657	1,868	+1,211
Full-corpus records	465,312	473,936	+8,624
Distinct threat-actor clusters	4	5	+ Miasma (new worm lineage)
Newest anchor event	IronWorm (Jun 2026)	AsyncAPI (14–15 Jul 2026)	signed-provenance malware

Three findings from the gap period drive the narrative for the rest of this report:

GAP FINDING 1

A new worm lineage — **Miasma** — emerged and adapted in real time. Six Miasma campaigns between 29 May and 26 June. Delivery mechanism pivoted from `postinstall` to `binding.gyp` within 72 hours of the first public write-ups. That is a directly observable adaptive response to defender detection.

GAP FINDING 2

Attackers now ship **validly-signed malware**. The AsyncAPI compromise (14 Jul) carries valid SLSA provenance because it was published through the repo's own npm OIDC release workflow after a stolen bot PAT. Miasma Wave 1 forges Sigstore provenance directly. Provenance attestation on its own is no longer a trust signal.

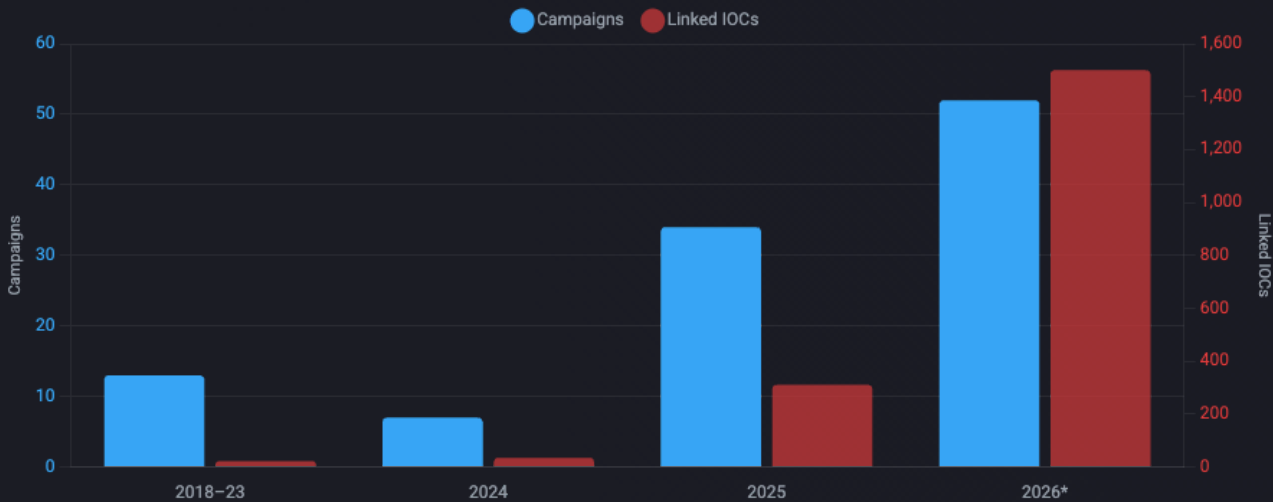
GAP FINDING 3

The corpus's most-corroborated high-trust namespace was hit again. The June baseline had named `@asyncapi/specs` as Shai-Hulud 2.0's patient zero (1.4M weekly, 100M+ lifetime downloads). Six weeks later the entire `@asyncapi` namespace was re-compromised via a GitHub Actions "pwn request": `@asyncapi/specs`, `@asyncapi/generator`, `@asyncapi/generator-helpers`, `@asyncapi/generator-components`. High-trust namespaces are being revisited, not retired.

The acceleration curve

The Phoenix Security Malware Package Intelligence corpus tracks 106 supply chain campaigns from 2018 through 15 July 2026, indexing 1,868 individual malicious package-versions as exact, scanner-ready IOCs. Plotting the corpus year by year, the curve does not look like growth. It looks like ignition, and it shows up at two different scales. The curated campaign layer surged into 2025; so did the full-corpus population layer. Both are still climbing into 2026.

SUPPLY-CHAIN CAMPAIGN ACCELERATION, 2024–2026 (CURATED CORPUS)

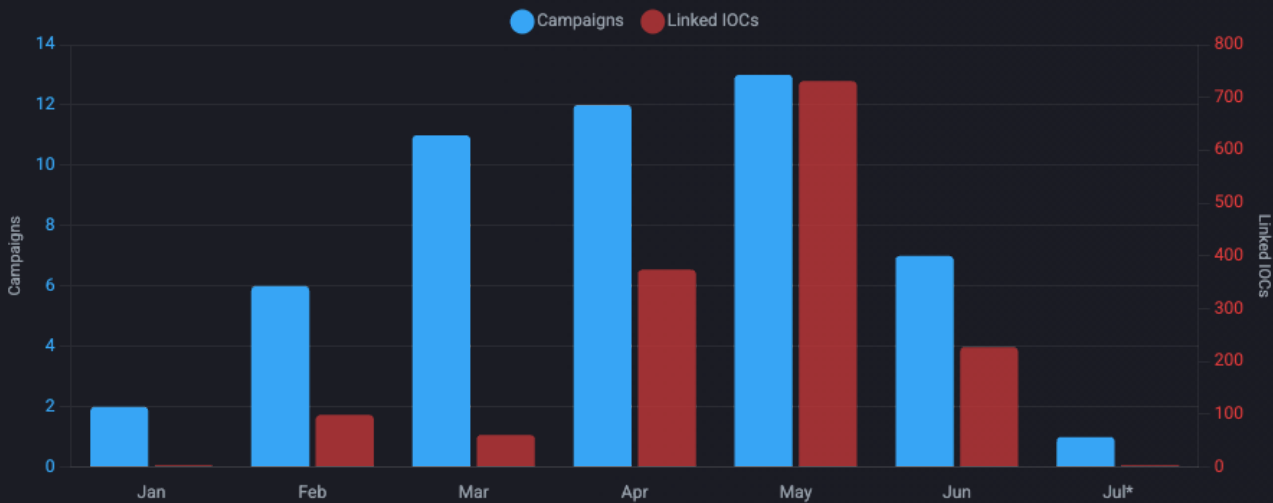


Curated tracker: 106 campaigns / 1,868 indexed IOCs across the full observation window. 2026 has already produced 1.5× the campaigns and roughly 4.8× the curated IOCs of all of 2025, and is not yet half elapsed by campaign-effort. 2025 itself was ~5× 2024.

YEAR OVER YEAR — TABULAR

YEAR	CAMPAIGNS	CURATED LINKED IOCS
2018–2023 (long tail)	13	22
2024	7	35
2025	34	311
2026 (to 15 Jul)	52	1,500

INSIDE 2026 — MONTH BY MONTH (TO 15 JULY)



May 2026 remains the peak (13 campaigns / 731 IOCs), driven by mass-propagation worms. June is a full-month figure now (7 / 227) versus the June-baseline partial "to the 7th" snapshot (4 / 74). The single July campaign is the AsyncAPI compromise: low package count, disproportionately high significance.

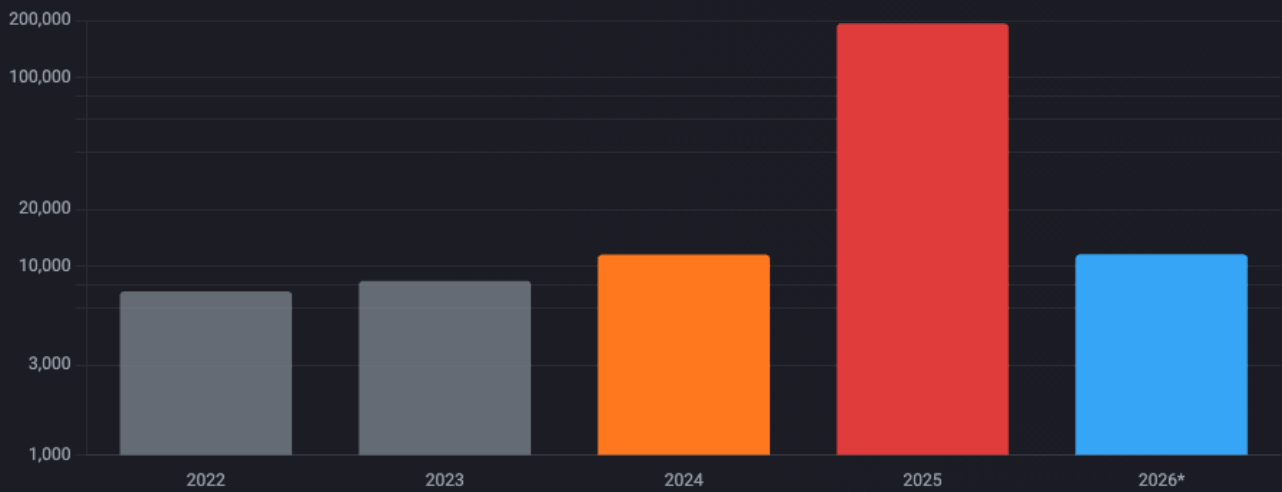
2026 MONTH BY MONTH — TABULAR

MONTH (2026)	NEW CAMPAIGNS	LINKED IOCS
Jan	2	4
Feb	6	99
Mar	11	61
Apr	12	374
May	13	731
Jun	7	227
Jul (to 15th)	1	4

Full-corpus incidents by year — the 17× surge in database context

The curated 1,868-IOC layer is the high-signal subset. Underneath it, the full `compromised_package_intel` corpus holds 473,936 records (97.6% OSSF bulk-feed-derived), of which 231,811 carry a dated incident. That fuller population tells the same story at a different scale:

COMPROMISED-PACKAGE INCIDENTS BY YEAR (FULL CORPUS, DATED RECORDS)



2025 produced 192,933 dated incidents, roughly 17× 2024 and larger than every prior year combined. 2026 is partial (to July) but has already surpassed the entire 2024 total. npm continues to drive essentially all of the growth.

FULL-CORPUS INCIDENTS — TABULAR

YEAR	COMPROMISED-PACKAGE INCIDENTS	YOY
2022	7,378	—
2023	8,385	+14%
2024	11,530	+37%
2025	192,933	+1,573% (≈17×)
2026 (to Jul)	11,572	partial (already > all of 2024)



Protect your developer machine from modern supply chain attacks today — for free. Get your Phoenix Blue Firewall Shield [here](#) →

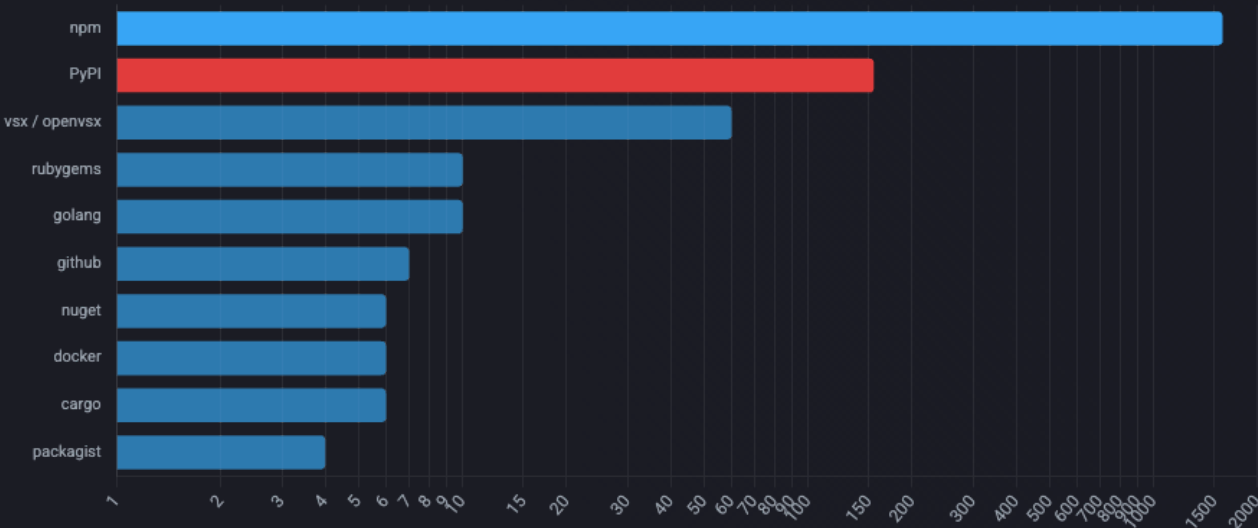
Ecosystem breakdown

npm's share of the curated tracker has widened, not narrowed, since the June baseline. It moved from 79% to 85% as the Miasma and IronWorm npm waves landed. PyPI stays the steady second front. The pattern lines up with the baseline thesis: Python attacks are lower-volume and higher-precision (RATs, credential stealers, banking trojans), Node attacks are worm-driven cascades. The full-corpus population picture, weighted by OSSF-bulk-feed volume, is even more npm-dominated: 93.2%.

CURATED CAMPAIGN IOCS BY ECOSYSTEM — 1,868 INDEXED PACKAGES (106 CAMPAIGNS)

ECOSYSTEM	LINKED IOCS	CAMPAIGNS	SHARE
npm	1,582	66	84.7%
PyPI	155	31	8.3%
vsx / openvsx	60	7	3.2%
rubygems	10	4	0.5%
golang	10	2	0.5%
github / github_actions	7	3	0.4%
nuget	6	1	0.3%
docker	6	3	0.3%
cargo	6	1	0.3%
packagist	4	1	0.2%

CURATED IOCS BY ECOSYSTEM — SHARE OF 1,868 INDEXED (LOG SCALE)



npm accounts for 84.7% of curated campaign IOCs (1,582 of 1,868) and appears in 66 of 106 campaigns. PyPI is the steady second front. A log scale keeps the minor ecosystems — RubyGems, Golang, NuGet, Cargo, Packagist — legible against npm's dominance.

What defenders should do — refreshed for the July gap

Six takeaways from the six-week gap between June and July. Every one is a direct response to a specific 2026 event, not a generic best-practice.

REFRESHED DEFENDER TAKEAWAYS — MAPPED TO JULY GAP EVENTS

#	TAKEAWAY	WHY (JULY GAP EVENT)
1	Provenance is necessary but no longer sufficient. Verify provenance and behaviour; treat a signed package from a compromised pipeline as the new default threat.	SLSA/Sigstore attestation can be forged (Miasma Wave 1) or legitimately produced over malicious code when CI is compromised upstream of signing (AsyncAPI).
2	--ignore-scripts is not a moat. Native-build steps and non-Node runtimes must be inside the monitoring boundary.	The <code>binding.gyp</code> / <code>node-gyp</code> and Bun-execution shift deliberately routes around lifecycle-script defences.
3	Harden CI/CD against "pwn requests". Audit every <code>pull_request_target</code> workflow that runs untrusted code with secret access. Scope bot PATs minimally and rotate on any anomaly.	AsyncAPI compromise was initiated through a <code>pull_request_target</code> workflow with repo secret access; the stolen bot PAT was the primary initial-access vector.
4	Assume AI-tooling config files are targets. Monitor <code>~/.claude/settings.json</code> , <code>.vscode/tasks.json</code> , and agent context files like credential stores.	Miasma Wave 2 and successors write persistence into these files; survives package uninstall.
5	Protect tokens — including AI-platform keys — as crown jewels. Short-lived, scoped, hardware-bound credentials break the worm loop.	Anthropic and OpenAI API keys now appear alongside npm/GitHub/cloud tokens in Miasma and IronWorm credential-sweep lists.
6	Watch the base of the graph, and watch it repeatedly. Transitive-dependency and namespace monitoring must be continuous, not one-shot.	High-trust namespaces are being re-targeted after cleanup. <code>@asyncapi/specs</code> was hit in Shai-Hulud 2.0 and again on 14 July.

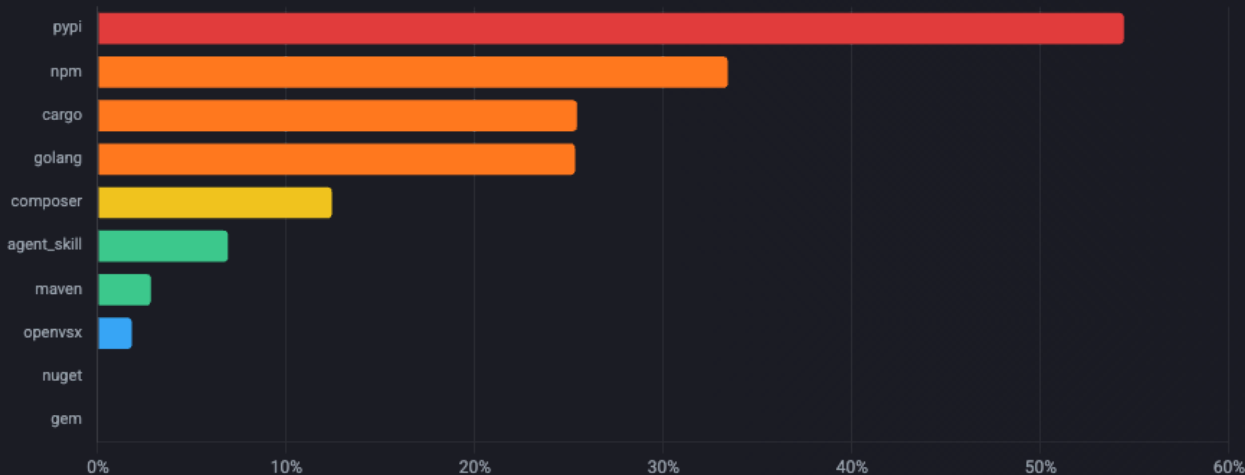
3-month broader scan — the live ecosystem picture

The 657-IOC campaign corpus tracks attributed waves — what attackers organised into named operations. The broader behavioural scan tracks the underlying population: every package the platform observed across registries over the last 90 days. The two views complement each other. Where the campaign data tells you which threat actor moved against which ecosystem, the population scan tells you the baseline behavioural risk rate across all of open source — and where the analysis frontier sits today.

Suspect and confirmed are treated as the same severity tier below. A suspect classification means the package triggered behavioural heuristics that warrant blocking; a confirmed one means manual verification has closed the loop. Both are "do not install" outcomes in production.

588,398 PACKAGES SCANNED (3 MONTHS)	32.7% SUSPECT + CONFIRMED RISKY (CROSS-ECOSYSTEM)	2,783 CONFIRMED MALICIOUS — 1 IN 211 OVERALL	1 in 90 NPM PACKAGES CONFIRMED MALICIOUS (HIGHEST CONCENTRATION)
---	--	---	--

RISK RATE BY ECOSYSTEM — % SUSPECT + CONFIRMED OF CLASSIFIED PACKAGES (3-MONTH SCAN)



Risk rate = (suspect + confirmed) ÷ (scanned - awaiting). Numbers in parentheses are confirmed-malicious counts. PyPI sits at the top of the population risk distribution despite carrying only 9% of campaign IOCs — the population scan catches a different signal than the campaign tracker.

PER-ECOSYSTEM BREAKDOWN — SCAN COUNTS AND RATIOS

ECOSYSTEM	SCANNED	CLEAN	SUSPECT	CONFIRMED	AWAITING	RISK RATE
pypi	223,213	90,226	119,146	1,339	1,800	54.4%
npm	90,872	55,691	29,336	1,006	146	33.4%
cargo	28,590	20,953	7,208	28	88	25.4%
golang	129,214	93,756	32,198	410	107	25.3%
composer	394	343	49	0	0	12.4%
agent_skill	26,213	21,706	1,613	0	2,894	6.9%
maven	143	139	4	0	0	2.8%
openvsx	33,806	10,720	194	0	22,892	1.8% <i>incomplete</i>
nuget	42,844	42,843	1	0	0	0.0%
gem	13,109	13,109	0	0	0	0.0%
TOTAL	588,398	349,486	189,749	2,783	27,927	34.4%

What the broader scan tells you that the campaign corpus does not

PyPI carries the highest behavioural risk rate at 54.4% — more than 1 in 2 PyPI packages scanned in the last 3 months triggered suspect or confirmed flags. This sits alongside the earlier ecosystem-vendor finding that PyPI campaign volume has declined since mandatory 2FA, and the two are reconcilable. Campaign-organised PyPI attacks are down, because attackers route around 2FA. But the population of single-shot, low-volume malicious PyPI packages — RATs, credential stealers, banking trojans — is now the largest risky-package population in the corpus. The campaign tracker undercounts PyPI; the behavioural scan does not.

npm has the highest confirmed-malicious concentration. 1,006 confirmed packages out of 90,872 scanned — roughly 1 in 90. That's an order of magnitude above any other ecosystem. The 33.4% suspect+confirmed rate aligns with the campaign-corpus finding that npm carries 79% of campaign IOCs. The two signals agree.

Cargo and Golang are dirtier than reputation suggests. Both sit at roughly 25% suspect+confirmed. Cargo has 28 confirmed malicious packages in 90 days; Golang has 410. Both ecosystems are under-represented in the campaign corpus (1 campaign each) because attackers haven't organised them into worm-style operations yet — but the behavioural baseline is already comparable to the registries that get headlines.

OpenVSX has 22,892 packages awaiting analysis — 68% of the scanned set. The 1.8% risk rate is computed against the 10,914 classified packages; the actual population risk could be materially higher once the awaiting backlog clears. Treat OpenVSX numbers as a lower bound only.

NuGet, gem, and maven are effectively clean at the population level (0%, 0%, 2.8%). All three carry mandatory 2FA or strong publisher friction. The campaign-corpus finding that registry controls reduce malicious volume by 40-60% is corroborated at the population scan layer.

agent_skill broader scan: 6.9% risk rate — lower than the 15.6% deep-scan figure published earlier in this report because those are different scans. The 15.6% number comes from the May 24th knostic deep-scan of 3,267 skills against the full critical-rule set. The 6.9% here is the broader population pass over 26,213 skills using the standard suspect-classifier. Both are right; they measure different things.

TOP RISK PACKAGE PER ECOSYSTEM — PACKAGE RISK ANALYTICS

For each ecosystem, the package carrying the highest aggregate risk score across all indexed versions. Risk score = vulnerable-version count × average heuristic. A single high-risk package with many trojanized versions outweighs a low-risk package with one trojanized version.

ECOSYSTEM	TOP RISK PACKAGE	VULN VERSIONS	CONFIRMED	AVG HEURISTIC	AGGREGATE RISK
golang	github.com/rstudio/rstudio	830	0	100.0	1,238.50
pypi	ouroboros-ai	344	0	100.0	533.80
npm	instar	38	5	100.0	96.35
cargo	http-smtp-rele	15	0	96.5	55.51
composer	alex-kudrya/adminix	4	0	100.0	40.80
agent_skill	apify-debug-bundle	1	0	100.0	36.45
maven	org.wso2.carbon.identity.framework/org.wso2.carbon.idp.mgt.ui	1	0	100.0	36.45
nuget	EBitSaaS.Report	1	0	100.0	36.45
openvsx	Angular Extension Pack	9	0	0.0	—

Three packages worth flagging: `rstudio/rstudio` (golang, 830 vulnerable versions at 100 heuristic) is the single highest aggregate-risk package in the corpus — a tampered Go module masquerading as RStudio's repo path. `ouroboros-ai` (pypi, 344 versions) — the AI-themed name pattern is consistent with the TanStack/Mistral/UiPath impersonation cluster: attackers follow install volume, and "AI" is where the volume is. `instar` (npm) is the only one with confirmed-malicious versions in the table — 5 out of 38. The other top-risk packages are all suspect-only, awaiting either takedown or manual confirmation.

The VS Code frontier — 0 to 7 campaigns in 18 months

VS CODE / OPENVSX CAMPAIGN CHRONOLOGY

Detections
27 → 105 (4x)

Oct 2024

VSCoDe Crypto/Zoom Impersonation

18 extensions · AES PowerShell dropper

Oct 2025

GlassWorm Wave 1 — Invisible Unicode

35,800 downloads · Solana C2 · self-propagating

Jan 2026

MaliciousCorgi — AI Chatbot Spyware

1.5M installs · full file contents exfiltrated

Feb 2026

OX Security VSCode CVEs

128M combined install exposure · 4 extensions

Mar 2026

TeamPCP CI/CD Siege

OpenVSX extensions inside 5-day attack chain

Apr 2026

GlassWorm Wave 3 — 73 Clones + Zig Dropper

Cross-IDE · 320+ artifacts since Dec 2025

May 2026 · CRITICAL

TeamPCP — GitHub Internal Breach

ReversingLabs: malicious VS Code extension detections grew from 27 in 2024 to 105 in the first 10 months of 2025 — nearly quadrupled. The GitHub breach via poisoned VS Code extension is the highest-value confirmed impact in the corpus.

AI agent skills and MCP — the dirtiest new surface

Beyond the package registries and the IDE extension marketplaces, the platform separately indexes and scans AI-agent skills. The numbers from a focused deep-scan run on 2026-05-24 are the clearest internal signal of the AI-shift thesis: skills carry a markedly higher risk rate than IDE extensions, and more than 1 in 4 deep-scanned skills triggered a critical-risk finding.

RISK RATE COMPARISON — AI-AGENT SKILLS VS IDE EXTENSIONS (KNOSTIC SCAN, PHOENIX MPI)

VS CODE / OPENVSX EXTENSIONS

6.9%

flagged unhealthy / risky of those scanned

~34,300 indexed
11,909 scanned
820 flagged

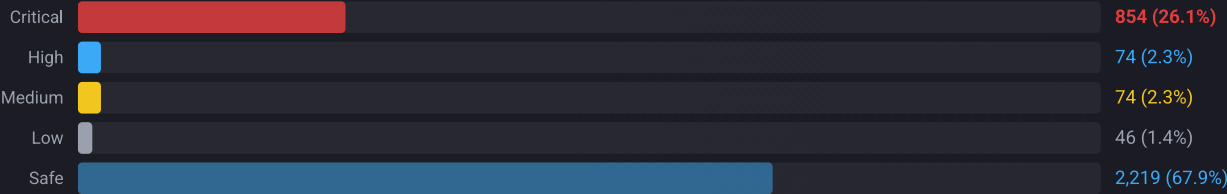
AI-AGENT SKILLS

15.6%

flagged unhealthy / risky of those scanned

~43,900 indexed
42,480 scanned
6,606 flagged

DEEP-SCAN RISK BREAKDOWN — 3,267 AI-AGENT SKILLS (2026-05-24)



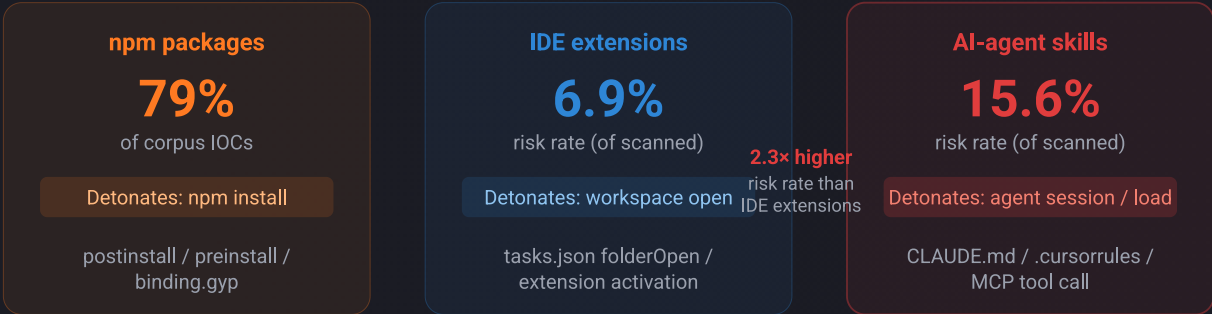
AI-AGENT SKILLS — DETECTION RULE HITS DRIVING CRITICAL FINDINGS

The most-triggered detection rules in critical-rated skills reveal a consistent pattern: shell command execution combined with network egress. These are skills that quietly run OS commands and phone home — the same primitive that defines postinstall-hook malware, now embedded in agent tooling. Popularity provides no safety signal: a `setup-deploy` skill in a 97,000-star repository appeared in the critical set.

DETECTION RULE	HITS	WHAT IT CATCHES
CS-009	3,603	Command / shell execution — skill invokes OS shell, subprocess, or eval
NS-002	2,075	Network egress — outbound connections to unconfigured external endpoints
RD-003	1,302	Remote data read — fetches remote instructions or payloads at runtime
NS-008	1,089	Suspicious network pattern — C2-consistent beacon or exfiltration pattern

The skills attack model: an AI coding assistant (Claude Code, Cursor, Copilot) loads a project-level skill or MCP server. That skill contains hidden instructions — delivered via zero-width Unicode (TrapDoor), a malicious `CLAUDE.md` entry (Miasma Wave 2), or a rogue MCP tool call (`SANDWORM_MODE`). The assistant executes the instruction as part of normal operation, exfiltrating secrets without the developer ever running a suspicious binary. The AI is not compromised — it is used as intended, against the developer.

THREE ATTACK SURFACES – RISK RATE AND DETONATION POINT



CONFIRMED CAMPAIGNS TARGETING AI AGENT TOOLING – SKILLS, MCP, AND CODING-ASSISTANT CONTEXT

CAMPAIGN	PERIOD	VECTOR	TECHNIQUE
SANDWORM_MODE	Feb 2026	MCP server injection	Git hook template abuse; AI MCP tool poisoning; Session P2P C2
TrapDoor	May 2026	Zero-width Unicode in .cursorrules / CLAUDE.md	Hidden instructions trick assistant into running "security scan" that exfiltrates secrets
Miasma Wave 2	Jun 2026	AI IDE config injection (Claude Code, Cursor, Gemini)	Drops backdoor configs into project dirs; binding.gyp detonation; future AI-generated code poisoned
GlassWorm (invisible Unicode)	Oct 2025–	OpenVSX extension → AI-agent context	Invisible-Unicode steganography born in VSX; technique jumped to AI-skill vector
Mini Shai-Hulud (SAP CAP wave)	Apr 2026	Claude Code SessionStart hook + VS Code tasks.json folderOpen	Persistence survives package uninstall; re-runs credential harvest on every session

External validation: Snyk's ToxicSkills research documents malicious AI-agent skills (ClawHub). Datadog's open dataset now formally tracks "AI Skills" and "IDE extensions" as first-class malware ecosystems — independent confirmation of this attack class.



Zero CVEs assigned during active exploitation. Across all 106 tracked campaigns. CVE-feed scanners had no detection surface for any campaign in this corpus. The attack surface is a trust assumption — a maintained package, a trusted publisher, a legitimate CI identity — not a code defect. No CVE gets issued because there is no flawed code path to assign one to.



Protect your developer machine from modern supply chain attacks today — for free. Get your Phoenix Blue Firewall Shield [here](#) →

Threat actor breakdown

Actor attribution is derived from campaign names and descriptions; the raw `threat_actor` column is populated for only 5 of 106 rows. Because a single campaign often references more than one lineage (a Miasma campaign will typically note its Shai-Hulud ancestry, for example), the counts below are *mention counts*, not disjoint totals. Treat them as cluster prominence rather than a partition.

THREAT-ACTOR CLUSTERS — CAMPAIGN MENTIONS AND DEFINING BEHAVIOUR

CLUSTER	CAMPAIGNS MENTIONING	MODEL	SIGNATURE
TeamPCP (UNC6780)	26	Automated, multi-ecosystem	CI/CD sieges; AI-tooling worm cascades (Trivy → Checkmarx → LiteLLM)
Shai-Hulud lineage	20	Self-replicating npm worms	Token theft, GitHub-repo exfiltration, automated reinfection
DPRK (Lazarus, Contagious Interview, Sapphire Sleet, Pungsan)	14	Social-engineering led	Fake recruiting → OtterCookie / BeaverTail / HexEval RATs
Miasma (NEW in July refresh)	6	Shai-Hulud derivative worm	binding.gyp delivery; Bun EDR-evasion; AI-tool persistence; provenance forging
SANDWORM_MODE	5	npm worm / MCP injection	Model Context Protocol context poisoning
IronWorm	1	Compiled-binary worm	Rust eBPF rootkit; OIDC self-propagation; Tor C2

Miasma — the gap period's defining new lineage

Miasma is a Shai-Hulud open-source derivative that produces unique per-infection encrypted payloads. Across six campaigns between 29 May and 26 June, it demonstrated a level of operational adaptivity not previously seen in the corpus. The delivery-vector pivot from `postinstall` to `binding.gyp` happened inside 72 hours of the first public write-ups. That is the clearest datapoint we have that these operators track defender reporting and re-tool inside a work-week.

MIASMA CLUSTER — SIX CAMPAIGNS, 29 MAY–26 JUNE

CAMPAIGN	FIRST SEEN	LINKED IOCS	DEFINING TRAIT
Wave 1 — Red Hat Cloud Services npm	29 May	14	Compromised Red Hat employee GitHub account; Sigstore provenance forging; GCP/Azure identity enumeration; EDR enumeration (CrowdStrike, SentinelOne, Carbon Black, StepSecurity)
Wave 2 — MIASMA_WAVE2_BINDINGYP (node-gyp)	1 Jun	68	Pivoted <code>postinstall</code> → <code>binding.gyp</code> within 72h of Wave-1 write-ups; 286 version-pinned IOCs; persistence survives <code>npm uninstall</code>
Wave 2 — Phantom Gyp / AI SDK	2 Jun	6	157-byte <code>binding.gyp</code> injected; bypasses <code>--ignore-scripts</code> ; Bun execution; drops backdoor configs for Claude Code, Cursor, Gemini
Hades Miasma — Mini Shai-Hulud PyPI worm	9 Jun	38	PyPI wheels with <code>*-setup.pth</code> startup execution; staged <code>_index.js</code> ; Bun download
Hades / LeoPlatform npm ATO	26 Jun	23	23 LeoPlatform packages republished under compromised maintainer; <code>binding.gyp <!(...) shell-exec</code> ; AES-128-GCM blob under Bun

Common Miasma TTPs: `binding.gyp` / `node-gyp` native-build execution instead of npm lifecycle scripts; Bun runtime download to execute payloads outside Node.js; AI coding-tool persistence via `~/.claude/settings.json`, `.vscode/tasks.json`, and `.github/workflows/codeql.yml` that survives package uninstall; credential sweeps that now explicitly target Anthropic and OpenAI API keys alongside npm/GitHub/cloud tokens; and Sigstore attestation forging on Wave 1. C2 rotates across Greek-mythology-themed infrastructure with Russian-locale detection causing silent exit.

AsyncAPI compromise (14–15 July 2026) — signed malware via a "pwn request"

The July anchor event, reported by Wiz and StepSecurity. Four malicious `@asyncai` npm packages (five versions) published on 14 July:

ATTACK CHAIN — ASYNCAPI NAMESPACE RE-COMPROMISE

STEP 1

Attacker exploits a `pull_request_target` workflow that runs untrusted PR code *with repo secrets*.

STEP 2

Steals the privileged `asynccapi-bot` PAT. Pushes commits to `asynccapi/generator` and `asynccapi/spec-json-schemas`.

STEP 3

Each repo's own release workflow publishes the trojanised packages through npm's GitHub OIDC integration. **Every malicious version therefore carries valid SLSA provenance.**

STEP 4

The `require()`-time dropper fetches a second stage from IPFS, delivering a Miasma-family RAT.

PACKAGE	MANAGER
<code>@asynccapi/specs</code>	npm
<code>@asynccapi/generator</code>	npm
<code>@asynccapi/generator-helpers</code>	npm
<code>@asynccapi/generator-components</code>	npm

Why this matters more than the small package count suggests: it is the corpus's clearest proof that provenance and attestation are not by themselves a trust boundary. The packages are signed, provenanced, and published by the legitimate CI pipeline. The compromise sits *upstream* of the signing step. And it re-hits `@asynccapi/specs`, the exact patient-zero the June baseline had identified for Shai-Hulud 2.0. High-trust namespaces are being revisited, not retired.

Key campaigns — May/June 2026 (densest 6 weeks on record)

MIASMA / SHAI-HULUD DERIVATIVE

Miasma Wave 2 — Phantom Gyp binding.gyp Bypass

Jun 2–3, 2026

57 packages

286 malicious versions

Under 2 hours

157-byte binding.gyp bypasses `--ignore-scripts` and all lifecycle monitors. Downloads Bun to evade Node.js EDR. AI IDE backdoors injected into Claude Code, Cursor, Gemini configs.

Full analysis on Phoenix Security blog →

UNKNOWN / NATION-STATE CAPABILITY LEVEL

IronWorm — Rust eBPF Rootkit, Tor C2, OIDC Self-Propagation

Jun 2026

37 packages

9 organizations

976 KB Rust binary

86 env vars swept. eBPF rootkit hides process. Tor C2. OIDC tokenless self-propagation. Sweeps all 2026 AI provider keys. Generational step from JS post-install scripts.

Full analysis on Phoenix Security blog →

UNKNOWN / SHAI-HULUD DERIVATIVE

Miasma Wave 1 — Red Hat @redhat-cloud-services npm

Jun 1, 2026

32+ packages

116,991 weekly DLs

GCP/Azure identity enumeration (not just static extraction). EDR checks CrowdStrike/SentinelOne pre-execution. Unique per-infection encrypted payloads. Sigstore provenance forgery.

[Full analysis on Phoenix Security blog →](#)

UNKNOWN

Megalodon — Mass GitHub Actions Backdoor (5,561 Repos)

May 18, 2026

5,718 commits

6 hours

CISA advisory

workflow_dispatch dormant backdoors activatable via GitHub API at any future time. Forged CI-bot identities with Sep 2001 timestamps. CISA issued advisory.

[Full analysis on Phoenix Security blog →](#)

UNKNOWN / DDJIDD564 CLUSTER

TrapDoor — AI Coding Assistant Poisoning via Zero-Width Unicode

May 19–22, 2026

34 packages

384 versions

3 ecosystems

npm + PyPI + Crates.io simultaneously. .cursorrules and CLAUDE.md injected with hidden Unicode instructions — AI assistant runs "security scan" that exfiltrates secrets. PRs to langchain, browser-use.

[Full analysis on Phoenix Security blog →](#)

UNKNOWN

Laravel-Lang Composer — 233 Versions, 700 Repos, 15 Minutes

May 22, 2026

Packagist/PHP

15 credential modules

GitHub fork tag-redirect — official repo appears clean. PHP autoload fires on every app boot. Self-deletes from disk. Targets 1Password, KeePass, HashiCorp Vault, AWS, CircleCI.

[Full analysis on Phoenix Security blog →](#)

Technique frequency and TTP shifts across 106 campaigns

Credential and token theft remains near-universal across the corpus. The July refresh adds three concrete TTP shifts worth calling out: (1) the majority of campaigns now target AI, LLM, or agent tooling in some form (56 of 106); (2) four campaigns deliver execution through binding.gyp / node-gyp native-build rather than lifecycle scripts, defeating --ignore-scripts; and (3) five campaigns download the Bun runtime specifically to sidestep Node-focused EDR process monitoring. Provenance forging or OIDC abuse now appears in two clusters (Miasma Wave 1 forges Sigstore attestation; AsyncAPI publishes with valid SLSA via a stolen bot PAT). All three shifts represent a deliberate migration away from the surfaces defenders spent 2024-2025 instrumenting.

TECHNIQUE FREQUENCY — CAMPAIGNS USING EACH TTP

TECHNIQUE	CAMPAIGNS	TREND
Credential / token theft (T1552)	~80	
Self-propagating worm (T1195.001)	27	
Crypto / wallet theft	20	
AI / LLM / agent tooling targeting	56	▲ majority case (56/106)
CI/CD compromise (T1195.002)	13	
Typosquatting (T1036.004)	13	
VS Code / extension abuse	12	
Maintainer account takeover	10	
Invisible / zero-width Unicode	6	▲ new in 2026
Install / postinstall lifecycle hooks	5	

73%

of indexed packages carry a pinned malicious version — precise, exact-version blockable IOCs

9 / 59

campaigns are cross-ecosystem — attackers no longer specialise in one registry (TrapDoor hit npm + PyPI + Crates.io simultaneously)

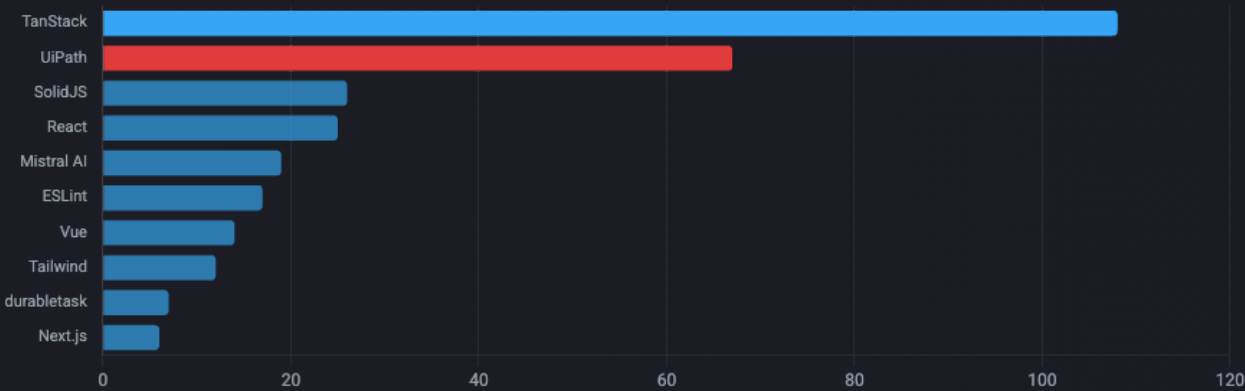


Protect your developer machine from modern supply chain attacks today — for free. Get your Phoenix Blue Firewall Shield [here →](#)

What attackers impersonate

The impersonation map confirms the AI shift. Modern front-end framework scopes and AI tooling now dominate, displacing the generic utility typosquatting that characterized earlier years.

BRAND / LIBRARY IMPERSONATION FREQUENCY IN MALICIOUS PACKAGE NAMES



TanStack (108) and UiPath (67) lead. AI tooling — Mistral, LiteLLM — clusters in the top tier alongside modern front-end frameworks. Attackers follow install volume.

New techniques introduced 2025–2026

Each technique in this table bypasses a previously recommended mitigation. The Phantom Gyp bypass makes the `--ignore-scripts` guidance from September 2025 structurally incomplete by June 2026.

NOVEL ATTACK TECHNIQUES AND WHAT THEY BYPASS

TECHNIQUE	CAMPAIGN	BYPASSES
Phantom Gyp (binding.gyp shell substitution)	Miasma Wave 2	All --ignore-scripts + lifecycle monitors
AI tool config poisoning (.cursorrules / CLAUDE.md)	TrapDoor, Miasma Wave 2	Developer review — AI assistant becomes exfiltrator
Python-to-JS execution bridge	PyTorch Lightning	PyPI signature verification (Python-only scope)
Bun runtime download	Mini Shai-Hulud, Miasma	Node.js process monitoring — EDR monitors node not bun
ICP canister C2	CanisterWorm, Namastex	Conventional domain takedown
GitHub fork tag-redirect	Laravel-Lang, actions-cool	Official repo appears clean in any code view
SLSA Level 3 / Sigstore provenance forgery	TeamPCP Wave 3, Miasma	Sigstore-based artifact verification
EDR enumeration before execution	Miasma Wave 1	Checks CrowdStrike / SentinelOne / Carbon Black pre-run
Unique per-infection encrypted payload	Miasma Wave 1	Version-hash fingerprinting and detection

ENRICHED MALWARE FAMILIES — TOP CATEGORIES IN HUMAN-ATTRIBUTED RECORDS

"Multi-stage AI toolchain poisoning" is the single most common enriched malware family in the corpus — the clearest internal signal of the AI-shift thesis. The near-even split between compromised legitimate packages and attacker-published packages confirms that maintainer account takeover is now as common as outright malicious publishing.

MALWARE FAMILY	RECORDS
Multi-stage AI toolchain poisoning	9
Browser-executed credential theft	7
Crypto-wallet hijackers	6
Scavenger (node-gyp.dll)	5
Cross-platform RAT	4
DanaBot / JarkaStealer (Java)	4
Linux backdoor (install-time)	3
TruffleHog + credential harvesting	2

Human-enriched records only. Phoenix campaign research layer.

COMPROMISED (TAKEOVER)

238,543

~51% of full corpus — legit packages taken over

MALICIOUS (ATTACKER-PUBLISHED)

226,701

~49% — packages authored and published by attackers

Account and maintainer compromise is now as common as outright malicious publishing. Scanning for new or unknown packages is not enough — the attack surface includes legitimate, established packages under new control.

KEY DETECTION IOCS — CONFIRMED ACROSS CORPUS

216.126.225.129:8443 — MEGALODON_CI C2
t.m-kosche[.]com — atool/AntV fake 0Tel collector C2
zero.masscan[.]cloud:443/v1/telemetry — Mini Shai-Hulud C2
flipboxstudio[.]info — Laravel-Lang C2
ddjidd564.github[.]io — TrapDoor PyPI remote payload
binding.gyp SHA-256: ef641e956f91d501b748085996303c96a64d67f63bfeef0dda175e5aa19cca90
Repo pattern: "Miasma: The Spreading Blight" OR reversed "niagA oG eW ereH :duLuH-iahS"
Commit identity: claude@users.noreply.github.com (spoofed automation)
GitHub search: Q0I9Imh0dHA6Ly8yMTYu (MEGALODON_CI injection marker)



Phoenix Blue Firewall Shield

Protect your developer machine from modern supply chain attacks today — for free.

Get the Shield →

PHOENIX SECURITY — VULNERABILITY RESEARCH

15 Jul 2026

[AsyncAPI Compromise: CI/CD Pwn Request Delivers Miasma RAT](#)

26 Jun 2026

[Miasma/Hades Hits LeoPlatform: binding.gyp Under Bun](#)

24 Jun 2026

[Close the Tap, Burn the Backlog: Agentic SDLC Security](#)

19 Jun 2026

[Mini Shai-Hulud Resurfaces: 1,600 Repos, 21 Accounts](#)

17 Jun 2026

[easy-day-js / Mastra: Typosquat RAT Across 144 npm Packages](#)

17 Jun 2026

[Phoenix Security Launches Blue Shield Supply Chain Firewall](#)

10 Jun 2026

[Miasma: The Worm That Lives Inside Your AI Tools](#)

10 Jun 2026

[Miasma Reaches Azure and PyPI: Hades Drops 37 Wheels](#)

8 Jun 2026

[The Acceleration: Supply Chain Attacks Go Industrial](#)

4 Jun 2026

[Miasma Is Back: binding.gyp Bypasses Postinstall Monitoring](#)

4 Jun 2026

[IronWorm: Rust npm Worm, eBPF Rootkit, Tor C2, 37 Packages](#)

2 Jun 2026

[Miasma: Red Hat Cloud Services npm Backdoored](#)

EXTERNAL REFERENCES

Anthropic · Jun 2026

[LLM ATT&CK Navigator: Mapping AI-Enabled Cyber Threats](#)

Sonatype · 2026

[2026 State of the Software Supply Chain](#)

ReversingLabs · 2026

[73% Rise in Malicious Open Source Packages](#)

Live IOC feed

[Phoenix MPI — 657 Indexed Malicious Package IOCs](#)



Phoenix Blue Firewall Shield

Protect your developer machine from modern supply chain attacks today — for free.

[Get the Shield →](#)

CVE scanners were blind to 100% of 106 documented campaigns

WHAT'S INSIDE

- 106 campaigns tracked across npm, PyPI, VS Code, and AI agent tooling
- 1,868 malicious package IOCs with exact version pinning
- Full threat actor profiles: TeamPCP, Shai-Hulud, Miasma, IronWorm, TrapDoor
- Detection commands and remediation guidance your team can use today



Scan to learn more