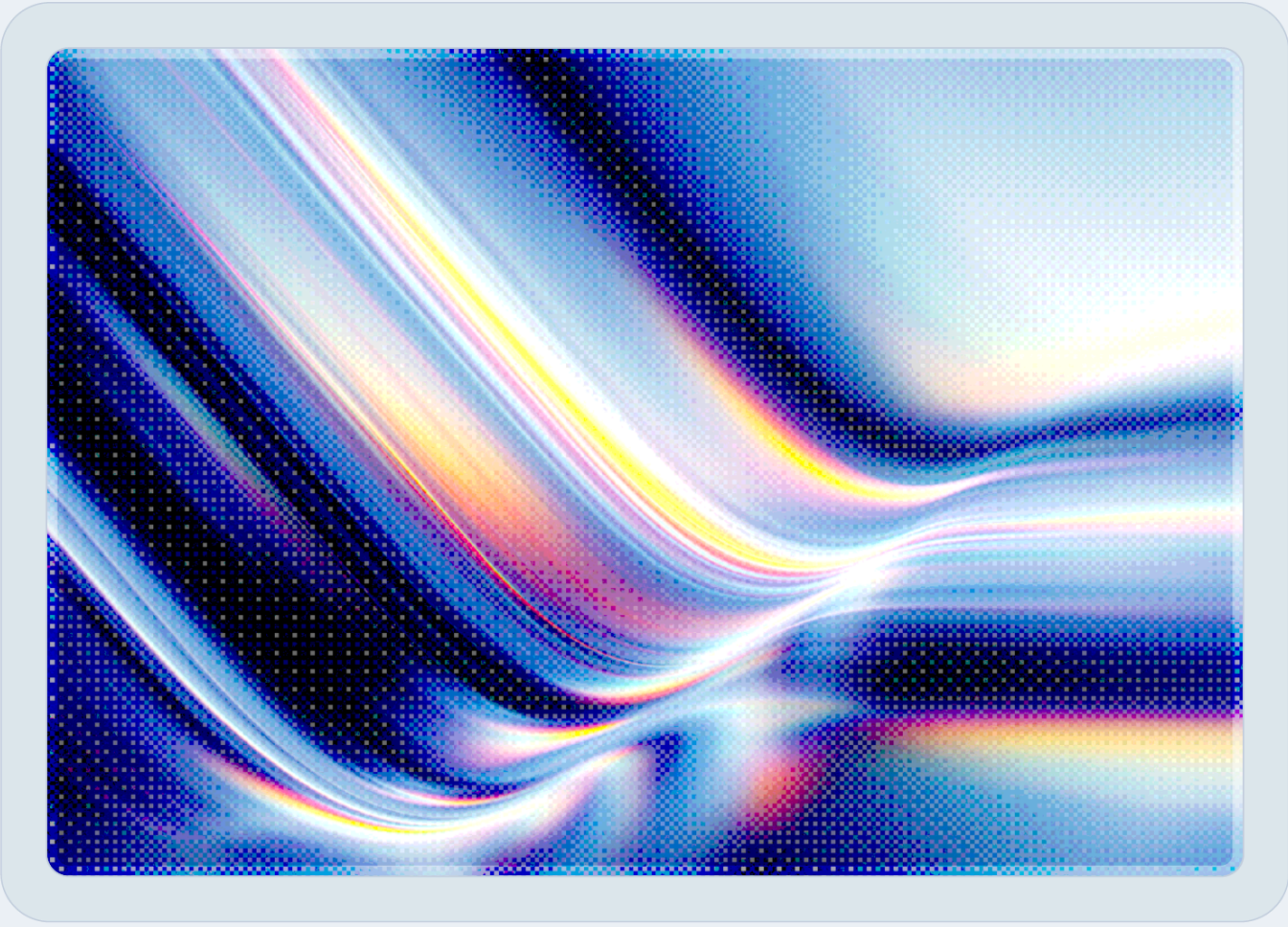
CSAI Foundation | Cloud Security Alliance

TeamPCP: Multi-Ecosystem Supply Chain Worm

2026-05-29

AI-assisted Rapid Research



CSAI

CSA cloud security alliance®

Key Takeaways

- TeamPCP (also tracked as DeadCatx3, PCPcat, ShellForce, and designated UNC6780 by Google Threat Intelligence Group) is responsible for the Mini Shai-Hulud campaign, which researchers have characterized as the first documented self-propagating supply chain worm to cross ecosystem boundaries autonomously, spreading across npm, PyPI, GitHub Actions, VS Code extensions, Docker Hub, and Kubernetes without requiring human operator involvement after initial deployment [4][20].
- The worm (CVE-2026-45321, CVSS 9.6) executed five distinct campaign waves between March and May 2026, compromising packages with a combined download footprint exceeding 518 million; Wave 3 alone resulted in approximately 373 malicious versions published across 171 npm and PyPI packages within five hours, and May 19, 2026 produced what Wiz Research and StepSecurity described as the largest single-hour supply chain event on record: more than 300 malicious versions across 323 packages [1][2][10].
- On May 18, 2026, a GitHub employee installed a trojanized VS Code extension (Nx Console, 2.2 million installs, verified publisher) that was live for approximately 11 minutes; TeamPCP subsequently exfiltrated approximately 3,800 GitHub internal repositories including github-mcp-server, github-oauth-proxy, and GitHub Sponsors infrastructure, marking the first documented supply chain attack that directly compromised platform infrastructure on which a large fraction of the global software development ecosystem depends [3][5][11].
- Wave 3 demonstrated that SLSA Build Level 3 provenance attestations are insufficient as a standalone defense: attackers extracted OIDC tokens from GitHub Actions runner memory via /proc/\<pid>/mem to generate valid attestations, and in later waves called Sigstore Fulcio and Rekor in real time to forge signing certificates, causing supply chain tooling to display green provenance badges on malicious packages [1][6][10].
- AI coding assistant configuration surfaces — Claude Code's ~/.claude/settings.json, Cursor, VS Code Continue, and Windsurf — were explicitly targeted by TeamPCP's worm and the related SANDWORM_MODE and TrapDoor campaigns, representing an emerging attack vector that enterprises deploying AI-assisted development must address as a first-class supply chain risk [22][23].

Background

The software supply chain attack surface has expanded significantly as developer toolchains have grown more interconnected. Each package registry, CI/CD platform, IDE extension marketplace, and AI coding assistant introduces trust relationships that are potential attack vectors. Security researchers had long theorized that a sufficiently sophisticated actor could chain these relationships into a self-propagating infection: compromise one package, harvest the credentials it exposes, use those credentials to compromise more packages, and repeat. The Mini Shai-Hulud worm — first deployed by TeamPCP in reconnaissance phases beginning September 2025, with large-scale public campaign waves launching in March 2026 — proved that this model was not merely theoretical [4][27].

TeamPCP's origins lie in cryptocurrency mining, theft, and ransomware [4]. The group's pivot to developer supply chain targeting suggests a structural insight: developer environments contain cloud provider keys, package registry tokens, SSH certificates, and CI/CD secrets that carry far more leverage than any individual consumer device. The worm's name, chosen by security researchers, references Frank Herbert's Dune sandworm — an organism that tunnels through entire ecosystems consuming everything in its path [4]. By the time the campaign's fifth documented wave concluded in late May 2026, TeamPCP had demonstrated capabilities that analysts expect will require years to fully incorporate into threat models and defensive architectures [4][10][27].

Several concurrent campaigns in this period — the Axios maintainer compromise in March 2026 [21], the SANDWORM_MODE MCP injection campaign in February 2026 [22], the Megalodon GitHub Actions mass-poisoning in May 2026 [19], and the GlassWorm VS Code campaign (October 2025–May 2026) [24][28] — collectively define a threat landscape in which developer toolchain trust surfaces have become the primary target class, not incidental collateral damage from campaigns aimed elsewhere.

Campaign Timeline and Technical Analysis

Early tooling and reconnaissance activity began September 2025; the five documented campaign waves targeting public packages launched between March and May 2026.

Wave 1: Security Tooling as an Entry Point (March 19–27, 2026)

TeamPCP's first major wave illustrated a strategic insight that would define the campaign: security tooling has privileged access to CI/CD pipelines by design. On March 19, 2026, attackers compromised Aqua Security Trivy's GitHub Actions workflow by exploiting mutable version tag pinning — a common

but dangerous practice in which workflow files reference an action by a floating version tag rather than a pinned commit SHA, allowing an attacker who has modified the referenced tag to inject arbitrary code into any pipeline that uses it [4][15][16]. From that initial foothold in Trivy's infrastructure, TeamPCP pivoted into Checkmarx KICS, LiteLLM, and the Telnyx Python SDK, affecting more than 10,000 CI/CD workflows [4][18].

The LiteLLM compromise had outsized exposure potential given the package's approximately 95 million monthly downloads [15]. Backdoored versions 1.82.7 and 1.82.8 were published to PyPI on March 24, 2026. The Telnyx Python SDK followed on March 27, 2026, with backdoored versions 4.87.1 and 4.87.2 [15]. In each case, the infection mechanism was the same: a compromised upstream CI/CD workflow extracted PyPI publish tokens from the victim's pipeline environment, and the worm used those tokens to push malicious package versions that contained its own propagation logic — ensuring that any developer who installed the affected packages would, in turn, expose their own publishing credentials [15][16][17].

The Axios maintainer account compromise on March 31, 2026, while attributed separately, followed the same pattern and prompted CISA Alert AA26-110A. Malicious versions axios@1.14.1 and axios@0.30.4 remained live for approximately three hours, a narrow window that nonetheless reflects the cascading speed at which package compromise can propagate through the npm ecosystem [21].

Wave 2: The CanisterWorm Variant (Late March 2026)

The second wave introduced the CanisterWorm variant, notable for its use of Internet Computer Protocol (ICP) canisters as command-and-control infrastructure. ICP canisters are smart contracts running on a decentralized blockchain, providing C2 channels that are resistant to conventional domain-blocking defenses — attackers are far more resistant to deplatforming from a decentralized blockchain than from a conventional C2 domain that can be sinkholed [20]. At least 64 npm packages and 140+ artifacts were affected, and the campaign exfiltrated approximately 300 gigabytes of data from an estimated 500,000 infected machines [4]. The use of blockchain-based C2 suggests TeamPCP was deliberately engineering around defensive countermeasures, not merely reusing commodity tooling.

Wave 3: Provenance Subversion at Scale (May 11, 2026)

Wave 3 represents the campaign's most technically consequential attack. StepSecurity AI Package Analyst detected the TanStack/Mistral AI attack on May 11, 2026; within five hours, approximately 373 malicious versions had been published across 171 npm and PyPI packages, including @tanstack/*, @mistralai/*, @opensearch-project/*, and UiPath packages with cumulative downloads exceeding 518 million [1][6][7][35].

The attack vector combined three distinct techniques. The pull_request_target workflow trigger, when misconfigured, executes with the base repository's secrets in the context of an attacker-controlled fork pull request — a pattern known as "Pwn Request" [1][6]. This was combined with GitHub Actions cache poisoning across the fork/base trust boundary. Most significantly, the attack extracted OIDC tokens from GitHub Actions runner process memory via /proc/\<pid>/mem — a technique that reads tokens directly from runner process address space, bypassing any credential storage protection [1][7]. The OIDC tokens obtained through this method were valid and short-lived, allowing them to be exchanged for npm publish credentials with full legitimacy as far as downstream verification tools were concerned.

The consequence for provenance-based defenses is severe and requires explicit acknowledgment. Wave 3 was the first documented case of malicious packages carrying valid SLSA Build Level 3 provenance attestations. In later waves, the worm called Sigstore Fulcio and Rekor at runtime to generate valid signing certificates for malicious packages, causing supply chain security tooling to display green provenance badges [10]. These techniques do not indicate a flaw in SLSA or Sigstore as frameworks; they indicate that process integrity controls cannot substitute for credential security. When the CI/CD process itself is compromised at the credential level, attestations of process integrity accurately describe a process that is producing malicious outputs. The SLSA framework remains a necessary control; Wave 3 demonstrates it is not sufficient in isolation [1][10][35].

CVE-2026-45321, with a CVSS score of 9.6, was assigned specifically for the TanStack compromise vector. No CVEs were assigned to the package compromise elements of the broader campaign, a gap that reflects the ongoing challenge of applying vulnerability taxonomy to supply chain attack techniques that exploit design properties of CI/CD systems rather than discrete software defects [10].

Wave 4: The GitHub Infrastructure Breach (May 18–20, 2026)

The fourth wave produced the campaign's most consequential single outcome. On May 18, 2026, attackers trojanized the Nx Console VS Code extension (nrwl.angular-console v18.95.0), which held 2.2 million installs and verified publisher status from Microsoft's marketplace [3][12]. The malicious version was live for approximately 11 minutes, from 12:36 to 12:47 UTC. In that window, a GitHub employee installed the extension. The malicious extension targeted a broad credential surface: GitHub tokens, npm tokens, AWS credentials, HashiCorp Vault tokens, Kubernetes service account tokens, and 1Password data. It also explicitly targeted Claude Code configuration files at ~/.claude/settings.json [3][12][37].

The GitHub breach, confirmed publicly on May 20, 2026, resulted in the exfiltration of approximately 3,800 internal repositories [5][11][37]. The stolen repositories included github-mcp-server, github-oauth-proxy, and GitHub Sponsors infrastructure — GitHub's own security tooling and monetization infrastructure. TeamPCP subsequently offered the stolen repositories for sale at $50,000 [5][11]. This

outcome distinguishes Wave 4 from prior waves: a supply chain attack that breached a platform operator's internal infrastructure, rather than affecting only downstream consumers, raises concerns about second-order attacks that cannot yet be fully characterized.

The same day, three backdoored versions of Microsoft's durabletask Python SDK (1.4.1, 1.4.2, 1.4.3; approximately 417,000 monthly downloads) were published to PyPI within a 35-minute window [13][14]. Endor Labs detected the compromise within two minutes of publication, a detection speed that likely limited downstream impact [3]. A notable operational security marker in the durabletask payload: the dropper skips systems with Russian locale settings, consistent with Eastern European cybercrime operational norms [13][14].

Concurrent with Wave 4, the Megalodon campaign used infostealer-harvested GitHub credentials to inject malicious workflows into 5,561 repositories across a six-hour window [19]. A significant portion of affected accounts correlated with known infostealer infection records, illustrating how compromised developer credentials from unrelated infostealers feed directly into supply chain attacks [19][36].

Wave 5: The @antv Namespace and Campaign Scale (May 19, 2026)

Wiz Research detected the fifth wave on May 19, 2026: the @antv npm namespace — an Alibaba-maintained data visualization ecosystem — was compromised alongside GitHub Actions workflows, with 'firedalazer' as the embedded C2 trigger phrase and malicious repositories marked with the reversed description 'niagA oG eW ereH :duluH-iahS' [2][9]. Wave 5 occurred on the same day as the durabletask compromise, and combined activity across both incidents produced what Wiz Research and StepSecurity described as the largest single-hour supply chain event on record: more than 300 malicious versions across 323 packages in 60 minutes [2][9].

Worm Mechanics, Persistence, and Ecosystem Dynamics

The Mini Shai-Hulud dropper (rope.pyz, 28 kilobytes) harvests credentials from 90 to 100+ developer tool paths spanning AWS, Azure, GCP, Kubernetes, 1Password, HashiCorp Vault, and AI tools including Claude Desktop and Cursor, then establishes persistence via hooks in .vscode/ and .claude/ directories and OS-level services (systemd on Linux, LaunchAgent on macOS) [3][39]. The FIRESCALE C2 mechanism routes commands through GitHub's public commit API with RSA-signed payloads, rotating C2 URLs in a way that cannot be blocked without disrupting legitimate development workflows [39]. The worm also deploys a destructive daemon, gh-token-monitor, that polls GitHub every 60 seconds and executes rm -rf ~/ if a stolen token is revoked — inverting the normal incident response step of credential revocation into a trigger for irreversible endpoint destruction [3][39].

On May 12, 2026, TeamPCP open-sourced the Shai-Hulud worm code under MIT license and posted a $1,000 BreachForums contest for the largest derivative attack, commoditizing the capability for actors with far less sophistication [39]. TeamPCP announced ransomware partnerships with the Vect group and Lapsus$ for credential monetization [39]. The campaign also attracted a competitor: SentinelLABS identified PCPJack on April 28, 2026, and published a technical analysis on May 7, 2026 — a rival worm that evicts TeamPCP tooling and installs its own harvester [8][38]. PCPJack's existence illustrates the emerging economics of developer environment compromise — infected machines are contested terrain. Hunt.io attributed all campaign waves to TeamPCP via overlapping IP ranges in the 83.142.209.0/24 subnet [39].

Security Analysis

The TeamPCP campaign's most significant analytical finding is that the defenses the industry converged on as best practices over the preceding three years are each individually insufficient. Pinning dependencies by version hash prevents typosquatting but not a legitimate version being replaced under the same number. SLSA Build Level 3 attestations prove a package was built through a specific process, not that the process was unsubverted at the credential level. Organizations that have implemented the full SLSA Build Level 3 requirements — including ephemeral, isolated build environments with no persistent credentials — would have constrained this attack vector; the failure mode documented in Wave 3 exploited partial implementations that met workflow requirements while retaining long-lived OIDC tokens accessible in process memory. Sigstore signing certificates prove identity but are available to anyone who can impersonate a legitimate CI/CD identity — including an attacker who has extracted an OIDC token from runner process memory. None of these controls are wrong; all are necessary; none is sufficient when the CI/CD process itself is compromised at the credential level.

The VS Code extension attack surface is structurally underestimated. Wiz Research documented 550+ validated leaked secrets across 500+ extensions in 2025, including 100+ cases where leaked marketplace tokens could push malicious updates to an extension's entire install base [26]. VS Code forks — Cursor, Windsurf, Google Antigravity — use the Open VSX registry, where unclaimed namespaces for extensions present in Microsoft's marketplace can be registered by any party, enabling squatting attacks without typosquatting [26]. The Nx Console compromise showed that an approximately 11-minute availability window for a verified, 2.2-million-install extension is sufficient to trigger a major infrastructure breach.

The convergence of AI coding assistant targeting across TeamPCP, SANDWORM_MODE, and TrapDoor reflects a structural vulnerability that most enterprises have not yet incorporated into supply chain security models. TeamPCP's worm explicitly harvests ~/.claude/settings.json; SANDWORM_MODE

injected rogue MCP servers into Claude Code, Cursor, VS Code Continue, and Windsurf configurations [22]; TrapDoor embedded hidden instructions in .cursorrules and CLAUDE.md context files to manipulate AI assistants into performing unauthorized credential scans [23]. As AI-assisted development becomes standard, these configuration surfaces are high-value attacker targets requiring explicit governance.

Recommendations

Immediate Actions

Organizations should treat any developer environment that installed affected packages between March and May 2026 as fully compromised. Mandatory credential rotation must cover npm and PyPI tokens, GitHub personal access tokens, AWS IAM credentials, GCP service account keys, Azure service principals, Kubernetes service account tokens, HashiCorp Vault tokens, and SSH keys for all affected developer accounts. Critically, because the gh-token-monitor destructive payload triggers on token revocation, organizations should isolate suspected-infected systems forensically before revoking credentials — revocation without isolation may trigger rm -rf ~/ on the infected endpoint. Password manager entries for affected developers should also be rotated given the worm's documented targeting of 1Password credential stores.

For CI/CD pipelines, all GitHub Actions workflow files should be audited for pull_request_target usage and all action references should be immediately converted from floating version tags to pinned full commit SHAs. npm install invocations in CI/CD environments should use the --ignore-scripts flag to prevent postinstall hook execution, eliminating the primary propagation mechanism.

Short-Term Mitigations

For ongoing npm and PyPI dependency management, organizations should pin dependencies using package-lock.json integrity hashes and, where private registry infrastructure is available, configure registry-side quarantine windows in products such as Artifactory or Sonatype Nexus to give detection tooling time to flag malicious updates before they execute in production pipelines. PyPI dependencies should be pinned to specific versions with cryptographic hashes in requirements files, and monitoring should alert on any unexpected new releases of pinned packages. Runtime monitoring of CI/CD service accounts should flag unusual npm publish or PyPI upload activity as a high-priority alert.

VS Code extension governance requires a structural response. Organizations should establish and enforce allowlists of approved extensions, monitor extension update events in endpoint security tooling, and treat extension installations in CI/CD environments with the same scrutiny as package installations. Extension marketplace credentials in developer environments should be subject to secret scanning. Given the GlassWorm campaign's demonstrated use of extensionPack and extensionDependencies manifest fields to chain trusted extensions into malware delivery, extension dependency chains should be reviewed, not just the extension itself [24][25][28].

For AI coding assistant security, MCP server configurations in Claude Code, Claude Desktop, Cursor, and comparable tools should be audited against known-good baselines. Approved MCP server allowlists should be established and enforced. Monitoring for unauthorized additions to ~/.claude/settings.json and equivalent paths should be incorporated into endpoint detection tooling. Project-level context files (.cursorrules, CLAUDE.md) sourced from external repositories should be treated with the same scrutiny as executable code, given their demonstrated ability to manipulate AI assistant behavior [22][23].

Strategic Considerations

The campaign's demonstration that SLSA Build Level 3 attestations can be forged when CI/CD credentials are compromised requires organizations to reframe how they communicate the assurance value of provenance controls to leadership and auditors. Provenance verification is a necessary control and organizations should continue pursuing SLSA adoption; however, it must be communicated as one layer of a defense-in-depth architecture, not a sufficient defense. Behavioral analysis of newly published packages and runtime monitoring of package installation side effects must complement provenance verification in mature supply chain security programs.

The open-sourcing of the Shai-Hulud worm code and the BreachForums derivative attack contest signal that this campaign's techniques will be replicated by actors with far less sophistication than TeamPCP. Organizations should plan their supply chain security investments against a threat model that assumes these capabilities are now broadly available, not limited to the original threat actor.

Regulatory deadlines are approaching. The European Commission's Cyber Resilience Act begins mandatory vulnerability reporting on September 11, 2026, and organizations subject to that regulation should ensure their supply chain incident response procedures include regulatory notification workflows [40]. NIST SSDF v1.2 is in public draft as of December 2025, incorporating lessons from recent supply chain incidents, and organizations should review it against their current secure development practices [31]. Generating and maintaining Software Bills of Materials for all internally developed and deployed software, consistent with the nineteen-nation SBOM shared vision guidance published by CISA and NSA on September 3, 2025, provides the visibility necessary to assess exposure when new campaign indicators are published [32].

CSA Resource Alignment

CSA's MAESTRO framework (Multi-layer AI Threat Enumeration and Risk Ontology) provides direct analytical coverage for the AI-specific attack vectors documented in this campaign. The SANDWORM_MODE MCP server injection, TeamPCP's targeting of Claude Code settings files, and TrapDoor's CLAUDE.md poisoning all represent attacks against the AI/ML supply chain at the software and third-party service layers of MAESTRO's model — the same layers addressed in the March 2026 NSA and allied nations' AI/ML Supply Chain Risks and Mitigations guidance that CSA Labs analyzed on March 17, 2026 [29][30].

CSA's AI Infrastructure Compliance Matrix (AICM), as a superset of the Cloud Controls Matrix, provides comprehensive organizational control mapping for response. The relevant AICM control domains — Supply Chain Management, Identity and Access Management, Infrastructure Security, and Incident Response — map directly to the attack patterns documented here: CI/CD credential harvesting, package registry account compromise, IDE extension trojanization, and AI assistant configuration tampering.

CSA's published guidance offers additional practitioner resources for organizations responding to this campaign. The April 2026 post on strengthening software supply chain security [33] and the September 2025 post on CI/CD per-run identities [34] address the root causes that TeamPCP exploited most directly. CSA Labs' prior research notes on GlassWorm [28] and Mini Shai-Hulud [27] provide organizational continuity across the related threat cluster.

References

[1] StepSecurity Research. "Mini Shai-Hulud Is Back: A Self-Spreading Supply Chain Attack Compromises TanStack npm Packages." StepSecurity, 2026-05-11.

[2] Wiz Research. "The Worm That Keeps on Digging: TeamPCP Hits @antv in Latest Wave." Wiz Research, 2026-05-19.

[3] Phoenix Security. "TeamPCP Wave Four: GitHub Breach via Poisoned VS Code Extension, durabletask PyPI Worm." Phoenix Security, 2026-05-20.

[4] Unit 42. "Weaponizing the Protectors: TeamPCP's Multi-Stage Supply Chain Attack on Security Infrastructure." Palo Alto Networks Unit 42, 2026-03-31.

[5] VentureBeat Security. "GitHub confirms 3,800 internal repos stolen through poisoned VS Code extension." VentureBeat, 2026-05-20.

[6] Orca Security. "TanStack and 160+ npm/PyPI Packages Compromised in Supply Chain Worm Attack." Orca Security, 2026-05-12.

[7] Expel Research. "Mini Shai Hulud: Cross-ecosystem supply chain worm targeting npm and PyPI." Expel, 2026-05-12.

[8] SentinelLABS. "PCPJack | Cloud Worm Evicts TeamPCP and Steals Credentials at Scale." SentinelOne SentinelLABS, 2026-05-07.

[9] StepSecurity Research. "Shai-Hulud: Here We Go Again – Mass npm Supply Chain Attack Hits the AntV Ecosystem." StepSecurity, 2026-05-19.

[10] Tenable Research. "Mini Shai-Hulud Supply Chain Attack CVE-2026-45321 FAQ." Tenable, 2026-05-12.

[11] BleepingComputer. "GitHub confirms breach of 3,800 repos via malicious VSCode extension." BleepingComputer, 2026-05-20.

[12] StepSecurity Research. "Nx Console VS Code Extension Compromised." StepSecurity, 2026-05-18.

[13] StepSecurity Research. "Microsoft's durabletask PyPI Package Compromised in Supply Chain Attack." StepSecurity, 2026-05-19.

[14] Wiz Research. "durabletask: TeamPCP's Latest PyPI Compromise." Wiz Research, 2026-05-19.

[15] Datadog Security Labs. "LiteLLM and Telnyx compromised on PyPI: Tracing the TeamPCP supply chain campaign." Datadog Security Labs, 2026-03.

[16] Snyk Security. "How a Poisoned Security Scanner Became the Key to Backdooring LiteLLM." Snyk, 2026-03.

[17] ReversingLabs Research. "The TeamPCP supply chain attack evolves." ReversingLabs, 2026-03-27.

[18] Endor Labs. "TeamPCP Isn't Done: Threat Actor Behind Trivy and KICS Compromises Now Hits LiteLLM." Endor Labs, 2026-03.

[19] StepSecurity Research. "Megalodon: Mass GitHub Actions Secret Exfiltration Across 5,500+ Public Repositories." StepSecurity, 2026-05.

[20] The Hacker News. "Self-Propagating Supply Chain Worm Hijacks npm Packages to Steal Developer Tokens." The Hacker News, 2026-04-22.

[21] CISA. "Supply Chain Compromise Impacts Axios Node Package Manager (CISA Alert AA26-110A)." CISA, 2026-04-20.

[22] Help Net Security. "Self-spreading npm malware targets developers in new supply chain attack (SANDWORM_MODE)." Help Net Security, 2026-02-24.

[23] The Hacker News. "TrapDoor Supply Chain Attack Spreads Credential-Stealing Malware via npm, PyPI, and CratesIO." The Hacker News, 2026-05-25.

[24] The Hacker News. "GlassWorm Supply-Chain Attack Abuses 72 Open VSX Extensions to Target Developers." The Hacker News, 2026-03.

[25] CrowdStrike Counter Adversary Operations. "Inside CrowdStrike's Takedown of a Developer-Targeting Botnet." CrowdStrike, 2026-05.

[26] Wiz Research. "Supply Chain Risk in VSCode Extension Marketplaces." Wiz Research, 2025.

[27] CSA Labs. "Mini Shai-Hulud: Multi-Ecosystem Developer Supply Chain Attack (CSA Labs Research Note)." CSA Labs, 2026-05.

[28] CSA Labs. "GlassWorm Returns: Developer Toolchain Worm Expands to GitHub and npm (CSA Labs Research Note)." CSA Labs, 2026.

[29] CSA Labs. "Eight-Nation AI/ML Supply Chain Risk and Mitigation Guidance (CSA Labs Research Note)." CSA Labs, 2026-03-17.

[30] NSA AI Security Center and allied agencies. "Artificial Intelligence and Machine Learning – Supply Chain Risks and Mitigations." NSA / defense.gov, 2026-03-04.

[31] NIST. "NIST SP 800-218 Rev. 1 SSDF v1.2 – Initial Public Draft." NIST CSRC, 2025-12-17.

[32] CISA. "CISA, NSA, and Global Partners Release Shared Vision of SBOM Guidance." CISA, 2025-09-03.

[33] CSA. "Strengthen Software Supply Chain Security (CSA Blog)." Cloud Security Alliance, 2026-04-21.

[34] CSA. "Securing CI/CD Pipelines with Per-Run Identities (CSA Blog)." Cloud Security Alliance, 2025-09-22.

[35] TanStack maintainers. "Postmortem: TanStack npm supply-chain compromise." TanStack, 2026-05.

[36] SecurityWeek. "Over 5,500 GitHub Repositories Infected in 'Megalodon' Supply Chain Attack." SecurityWeek, 2026-05.

[37] Help Net Security. "TeamPCP breached GitHub's internal codebase via poisoned VS Code extension." Help Net Security, 2026-05-20.

[38] BleepingComputer. "New PCPJack worm steals credentials, cleans TeamPCP infections." BleepingComputer, 2026-05-07.

[39] Phoenix Security. "Sha1-Hulud / Shai-Hulud: Full Technical Dissection of TeamPCP's Self-Propagating Supply Chain Worm." Phoenix Security, 2026-05.

[40] European Parliament and Council of the European Union. "Regulation (EU) 2024/2847 – Cyber Resilience Act." Official Journal of the European Union, 2024-10-23.

