Hive Pro

Threat Level

Red

Hiveforce Labs

THREAT ADVISORY

ATTACK REPORT

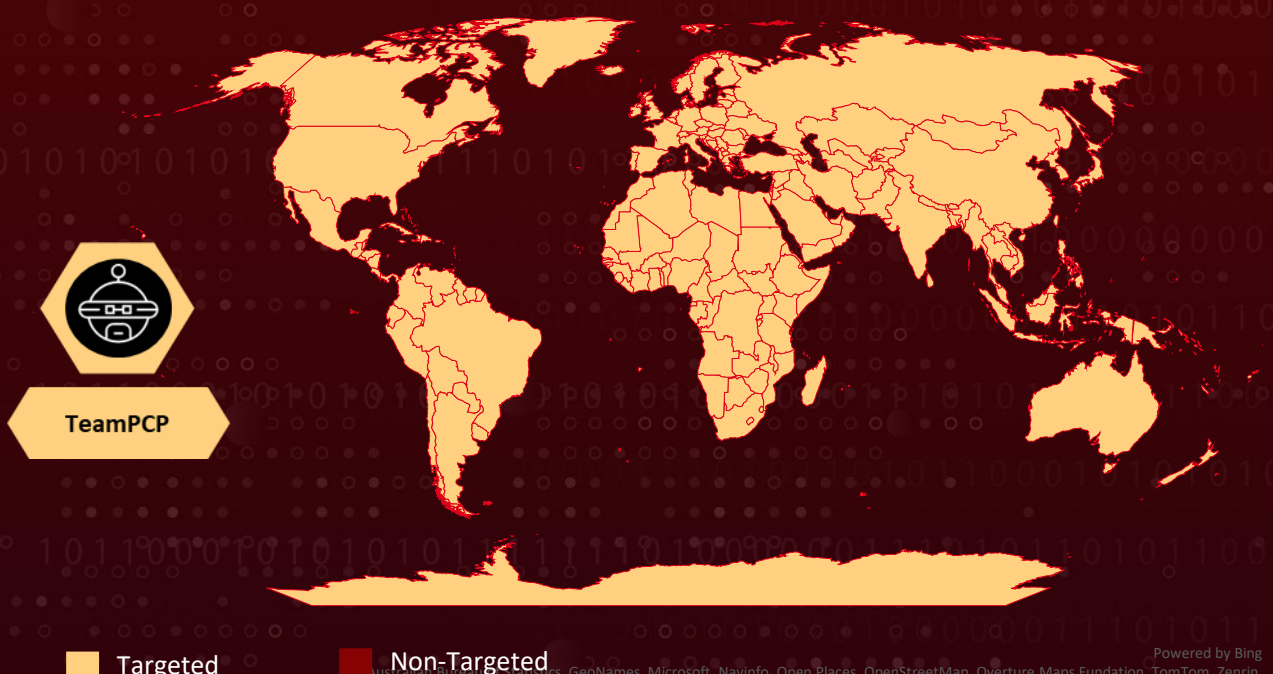
Mini Shai-Hulud npm Supply Chain Worm: TanStack & Multi-Ecosystem Compromise

Date of Publication	Last Updated Date	Admiralty Code	TA Number
May 15, 2026	May 19, 2026	A1	TA2026131

Summary

First Seen: May 10, 2026
Targeted Regions: Global (excludes systems configured with Russian language locale)
Targeted Platforms: npm (Node.js), PyPI (Python), GitHub Actions, macOS, Linux
Targeted Products: @tanstack/* packages (React Router, React Start, Vue Router, Solid Router, and related libraries), @uipath/* packages, @mistralai/mistralai, @opensearch-project/opensearch, guardrails-ai, intercom-client, Copycat packages: chalk-tempalte, axois-utils, @deadcode09284814/axios-util, color-style-utils,
Targeted Industries: Software Development, Enterprise Automation, Artificial Intelligence, Cloud Computing, CI/CD Pipelines, Data Analytics/Visualization
Threat Actor: TeamPCP (aka PCPcat, ShellForce, DeadCatx3, CipherForce, Persy_PCP, UNC6780)
Malware: Mini Shai-Hulud Worm
Attack: A coordinated supply chain attack by TeamPCP leveraging a chained exploit of GitHub Actions misconfigurations, including the pull_request_target "Pwn Request" pattern, CI cache poisoning across fork-to-base trust boundaries, and OIDC token extraction from runner process memory, to publish 84 malicious versions across 42 @tanstack/* npm packages. The worm-like malware harvests credentials from cloud providers, CI/CD systems, and developer environments, exfiltrates stolen data through multiple redundant channels, and self-propagates by republishing infected versions of other packages the victim maintains. TeamPCP subsequently open-sourced the worm code and launched an underground bounty challenge, spawning copycat variants with independent C2 infrastructure. A new wave on May 19 compromised 639 versions across 323 @antv ecosystem packages via a hijacked maintainer account, bringing the total campaign footprint to 1,055 malicious versions across 502 unique packages.

Attack Regions



CVE

CVE	NAME	AFFECTED PRODUCT	ZERO-DAY	CISA KEV	PATCH
CVE-2026-45321	TanStack Router npm Packages Embedded Malicious Code Vulnerability	TanStack Router npm Packages	✗	✗	✓

Attack Details

#1

On 11 May 2026, a coordinated supply-chain attack compromised 84 malicious versions across 42 @tanstack/* npm packages along with packages in the @uipath, @mistralai, @squawk, and @opensearch-project namespaces, plus PyPI's guardrails-ai and mistralai, over 170 affected packages total. With @tanstack/react-router at ~12 million weekly downloads, the blast radius is significant.

#2

Security researchers assess with high confidence that the campaign is operated by the threat group TeamPCP (also tracked as DeadCatx3, PCPcat, ShellForce, CipherForce), a financially motivated group that rose to prominence in late 2025. The attribution is supported by shared toolchain artifacts, including a verbatim ctf-scramble-v2 Fisher-Yates PRNG seeded with 0x3039 across the Bitwarden CLI, SAP, and TanStack payloads, identical C2 infrastructure, the malware's Russian-language locale check (payload terminates without exfiltrating if the system language is Russian), and the group's public claim of responsibility.

#3

The chain began on 10 May with a renamed fork of TanStack/router (zblgg/configuration) containing a malicious commit under a fabricated identity. A pull request opened the next day triggered bundle-size.yml, which used the pull_request_target "Pwn Request" pattern, executing fork-controlled code in the trusted base-repo context. That execution wrote a poisoned 1.1 GB pnpm-store entry into the GitHub Actions cache, keyed so release.yml would later restore it on the next push to main. The PR was then force-pushed to a no-op and closed.

#4

Detonation occurred when unrelated maintainer merges triggered release.yml. The poisoned cache restored, attacker binaries read /proc/<pid>/mem of the Runner.Worker process to extract the OIDC token minted for npm trusted publishing, then POSTed publishes directly to registry.npmjs.org, producing tarballs with valid SLSA Build Level 3 provenance. This is significant because SLSA provenance attests which pipeline produced the artifact, not whether the pipeline was behaving as intended, a compromised build step produces a validly-attested but malicious package, rendering provenance-based defences ineffective against this class of attack.

#5

The downstream payload is a credential stealer and self-propagating worm, exfiltrating cloud, Kubernetes, Vault, GitHub, npm, and SSH credentials over triple-redundant C2 (typosquat domain, Session messenger P2P, GitHub dead drops). It republishes other packages each victim maintains, driving lateral spread. The PyPI variant (guardrails-ai, mistralai) operates differently, a lightweight 13-line loader fetches a remote payload that for the first time targets password vaults (1Password, Bitwarden), and carries a destructive wiper targeting systems with Israeli or Iranian locale settings. Detection occurred ~20 minutes after the first publish, but malicious tarballs remained installable for hours due to npm's "no unpublish if dependents exist" policy, the actual exposure window extends well beyond detection.

#6

Critically, the payload installs a gh-token-monitor daemon that runs rm -rf ~/ on detecting GitHub token revocation. Additionally, the malware persists into .claude/ and .vscode/ directories as router_runtime.js or setup.mjs, these files survive npm uninstall, so package removal alone is not sufficient remediation. Responders must isolate or image affected hosts, remove both the daemon and IDE persistence artifacts, and only then rotate tokens, remediation otherwise triggers data destruction.

#7

TeamPCP open-sourced the complete Shai-Hulud worm code on GitHub with deployment instructions, using future-dated commits (January 1, 2099) to confuse forensics. The code hooks into AI-assisted coding tools to auto-execute on startup. They simultaneously launched a $1,000 bounty on a prominent underground forum for the biggest supply chain attack using their code. Copycat variants emerged within days, typosquatted packages deploying Shai-Hulud clones with independent C2 servers, including one converting victims into a DDoS botnet. Defenders now face a growing population of variants with different C2 endpoints and payloads sharing enough DNA to be dangerous but not enough to share detection signatures.

#8

On May 19, 2026, the compromised npm account "atool" was used to publish 639 malicious versions across 323 packages in a 22-minute burst, hitting widely used @antv data visualization libraries with millions of monthly downloads, plus a VS Code extension and the GitHub Action actions-cool/issues-helper. The obfuscated Bun payload harvests 20+ credential types and exfiltrates to t.m-kosche[.]com:443 disguised as an OpenTelemetry endpoint, with GitHub API dead drops as fallback, over 2,200 repos created with stolen tokens bearing the reversed marker "niagA oG eW ereH :duluH-iahS". The full campaign now totals 1,055 malicious versions across 502 unique packages. Separately, a major AI company confirmed two employee devices were compromised in the original TanStack wave, with application signing keys across all platforms impacted and mandatory user updates issued.

Recommendations

Disable the gh-token-monitor Persistence Mechanism Before Revoking Tokens: Before rotating any GitHub tokens, search all potentially affected developer machines and CI runners for the gh-token-monitor daemon files (~/.config/systemd/user/gh-token-monitor.service on Linux, ~/Library/LaunchAgents/com.user.gh-token-monitor.plist on macOS) and remove them. The malware's dead man's switch triggers a destructive home directory wipe (rm -rf ~/) upon detecting token revocation, so persistence removal must precede credential rotation.

Monitor for Self-Propagation Indicators: Watch npm publish logs for unexpected version bumps of exactly +3 patches with no corresponding changelog entry. This is the fingerprint of the Mini Shai-Hulud worm's automated propagation mechanism and indicates an infected package has been used to compromise additional packages.

Pin GitHub Actions to Commit SHAs: Replace all tag-based action references (e.g., actions/checkout@v6.0.2) with full commit SHA pins to prevent tag retargeting attacks, which carry the same blast radius as cache poisoning.

Rotate All Credentials on Exposed Hosts: If any affected package version was installed on a machine or CI runner, rotate AWS access keys, GCP service account keys, Azure service principals, HashiCorp Vault tokens, Kubernetes service account tokens, GitHub personal access tokens, npm publish tokens, and SSH private keys. For GitHub Actions environments specifically, assume all repository, organization, and environment secrets were extracted from runner memory regardless of workflow configuration.

Block C2 Infrastructure at Network Perimeter: Block outbound connections to git-tanstack[.]com, api.masscan[.]cloud, and .getsession.org (including filev2.getsession.org, seed1.getsession.org, seed2.getsession.org, seed3.getsession.org) at the DNS and proxy level. Also block 83.142.209[.]194, the IP address used by the Python variant's credential stealer.

Audit GitHub Repositories for Injected Workflows and Branch Poisoning: Search all organizational repositories for GitHub Actions workflows containing api.masscan[.]cloud, commits authored by "claude \<claude@users.noreply.github.com\>" with the message "chore: update dependencies" that add .claude/ or .vscode/ directories, and newly created repositories matching Dune-themed naming patterns (e.g., sardaukar-ornithopter-42).

Pin Dependencies and Disable Lifecycle Scripts in CI: Pin all @tanstack/ dependencies to the last confirmed clean versions (the x.x.65 releases) using lockfiles until the TanStack team publishes verified clean releases. Run npm install with --ignore-scripts in CI pipelines that do not require lifecycle script execution to prevent payload detonation during package installation.

Remove pull_request_target Workflows or Restrict Their Scope: Audit all GitHub Actions workflow files across organizational repositories for the pull_request_target trigger event. Workflows using this trigger must never check out or execute code from the pull request head. If base-repo permissions are needed to react to a PR, use the workflow_run pattern against artifacts from a sandboxed pull_request job, as recommended by GitHub's security team.

Potential MITRE ATT&CK TTPs

Tactic	Technique	Sub-technique
Initial Access	T1195: Supply Chain Compromise	T1195.002: Compromise Software Supply Chain
		T1195.001: Compromise Software Dependencies and Development Tools
Execution	T1204: User Execution	T1204.002: Malicious File
	T1059: Command and Scripting Interpreter	T1059.007: JavaScript
		T1059.006: Python
Persistence	T1543: Create or Modify System Process	T1543.001: Launch Agent
		T1543.002: Systemd Service
	T1546: Event Triggered Execution	
Defense Evasion	T1027: Obfuscated Files or Information	
	T1036: Masquerading	T1036.005: Match Legitimate Name or Location
	T1070: Indicator Removal	T1070.006: Timestomp
Credential Access	T1552: Unsecured Credentials	T1552.001: Credentials In Files
		T1552.005: Cloud Instance Metadata API
	T1528: Steal Application Access Token	
Discovery	T1526: Cloud Service Discovery	
Collection	T1005: Data from Local System	

Tactic	Technique	Sub-technique
Exfiltration	T1567: Exfiltration Over Web Service	T1567.001: Exfiltration to Code Repository
	T1041: Exfiltration Over C2 Channel	
Impact	T1485: Data Destruction	
Lateral Movement	T1072: Software Deployment Tools	
Command & Control	T1071: Application Layer Protocol	T1071.001: Web Protocols
	T1102: Web Service	T1102.002: Bidirectional Communication

Indicators of Compromise (IOCs)

TYPE	VALUE
Domains	git-tanstack[.]com, Api[.]masscan[.]cloud, t.m-kosche[.]com, 87e0bbc636999b[.]lhr[.]life
URLs	hxxp[:]//git-tanstack[.]com[:]443/router, hxxp[:]//api[.]masscan[.]cloud/v2/upload, hxxp[:]//litter[.]catbox[.]moe/h8nc9u[.]js, hxxp[:]//litter[.]catbox[.]moe/7rrc6l[.]mjs, hxxp[:]//git-tanstack[.]com/tmp/transformers[.]pyz, hxxp[:]//83[.]142[.]209[.]194/v1/weights, hxxp[:]//83[.]142[.]209[.]194/v1/models, hxxp[:]//83[.]142[.]209[.]194/audio[.]mp3
IPv4	83[.]142[.]209[.]194
IPv4:PORT	80[.]200[.]28[.]28[:]2222
Session Seed Node	seed1[.]getsession[.]org, seed2[.]getsession[.]org, seed3[.]getsession[.]org
Session File Server	filev2[.]getsession[.]org
Session Recipient ID	05f9e609d79eed391015e11380dee4b5c9ead0b6e2e7f0134e6e51767a87323026

TYPE	VALUE
Git Dependency (Dropper)	github:tanstack/router#79ac49eedf774dd4b0cfa308722bc463cfe5885c, github:antvis/G2#1916faa365f2788b6e193514872d51a242876569
GitHub Account	zblgg (ID: 127806521), voicproducoes (ID: 269549300) Deadcode09284814 atool
Forged Commit Identity	claude \<claude@users.noreply.github.com\>
Attacker Fork	github.com/zblgg/configuration
Malicious Workflow Run	github.com/TanStack/router/actions/runs/25613093674, github.com/TanStack/router/actions/runs/25691781302
SHA256	2ec78d556d696e208927cc503d48e4b5eb56b31abc2870c2ed2e98d6be27fc96, ab4fcadaec49c03278063dd269ea5eef82d24f2124a8e15d7b90f2fa8601266c, 2258284d65f63829bd67eaba01ef6f1ada2f593f9bbe41678b2df360bd90d3df, 1e8538c6e0563d50da0f2e097e979ebd5294ce1defe01d0b9fe361ba3bed1898, a68dd1e6a6e35ec3771e1f94fe796f55dfe65a2b94560516ff4ac189390dfa1c
SHA1	E7d582b98ca80690883175470e96f703ef6dc497, 12f35b1081b17d21815b35feb57ab03d02482116, 820fa07a7328b6cf2b417078e103721d4d8f2e79
Cache Key	Linux-pnpm-store-6f9233a50def742c09fde54f56553d6b449a535adf87d4083690539f49ae4da11
File Path	/tmp/tmp.ts018051808.lock, ~/.config/systemd/user/gh-token-monitor.service, ~/Library/LaunchAgents/com.user.gh-token-monitor.plist, .claude/setup.mjs, .vscode/setup.mjs, index.js
Commit Message Marker	IfYouRevokeThisTokenItWillWipeTheComputerOfTheOwner

Patch Links

https://github.com/TanStack/router/releases

https://github.com/antvis/g2/releases

References

https://tanstack.com/blog/incident-followup

https://tanstack.com/blog/npm-supply-chain-compromise-postmortem

https://www.wiz.io/blog/mini-shai-hulud-strikes-again-tanstack-more-npm-packages-compromised

https://www.upwind.io/feed/shai-hulud-tanstack-supply-chain-worm

https://hivepro.com/threat-advisory/teampcp-automated-supply-chain-from-trivy-to-litellm-in-a-multi-ecosystem-breach/

https://socket.dev/blog/antv-packages-compromised

https://safedep.io/mini-shai-hulud-strikes-again-314-npm-packages-compromised/

https://www.ox.security/blog/new-actors-deploy-shai-hulud-clones-teampcp-copycats-are-here/

What Next?

At **Hive Pro**, it is our mission to detect the most likely threats to your organization and to help you prevent them from happening.

Book a Demo of **HivePro**.

REPORT GENERATED ON

May 15, 2026 • 09:00 AM

© 2026 All Rights are Reserved by Hive Pro

More at www.hivepro.com