

The Mini Shai-Hulud Worm and the New Era of CI/CD Exploitation

By Flashpoint

Published: 2026-05-28 · Archived: 2026-08-06 02:02:06 UTC

The developer ecosystem recently faced one of its most significant architectural threats to date, with the [threat actor](#) group TeamPCP unleashing Mini Shai-Hulud—a self propagating [worm](#) and multi-ecosystem threat. Potentially affecting millions of developers and thousands of companies, Mini Shai-Hulud has fundamentally compromised the trust layer of modern CI/CD pipelines.

The operational tempo of Mini Shai-Hulud has accelerated with every campaign. What began as opportunistic credential theft has now evolved into a high-speed, automated operation that can compromise hundreds of packages in under thirty minutes. From the exfiltration of approximately 3,800 internal GitHub repositories to the poisoning of critical libraries like TanStack and AntV, TeamPCP's campaign has been incredibly effective in exploiting developer tooling and identity infrastructure.

What is Mini Shai-Hulud?

Mini Shai-Hulud is deployed as a 498 KB obfuscated script executed using the Bun JavaScript runtime. The deliberate choice of Bun, rather than Node.js, is a tactical evasion technique as most endpoint detection and response (EDR) platforms and security information and event management (SIEM) solutions have behavioral rules tuned to Node.js execution patterns.

How Mini Shai-Hulud Works

The worm propagates by stealing npm and GitHub authentication (OIDC) tokens from developer environments, then using those credentials to publish malicious versions of packages the compromised user maintains. To accomplish this, the worm scrapes runner process memory to extract short-lived identity tokens, which it then exchanges for per-package npm trusted-publisher tokens without requiring any long-lived npm secrets.

Credential Exfiltration and Command-and-Control

Mini Shai-Hulud targets credentials across 130 file paths, including npm tokens, GitHub personal access tokens, AWS, GCP, and Azure configuration files, Kubernetes kubeconfig files, Docker credentials, HashiCorp Vault tokens, 1Password and Bitwarden CLI vaults, SSH private keys, and Bitcoin wallet files.

Exfiltration occurs across multiple channels: the Session Protocol network, the GitHub Git Data API using dynamically created Dune-themed repositories on victim accounts, HTTPS to the threat actor-controlled domain, and an api for GitHub Actions workflow exfiltration.

The worm uses a dead-drop command-and-control (C2) architecture via GitHub's public commit search API. An installed daemon (kitty-monitor, deployed as a systemd service on Linux or a LaunchAgent on macOS) polls GitHub for commits containing the string "firedalazer," parses RSA-PSS-signed command payloads from matching commits, and executes them. This technique leverages GitHub as a trusted relay, making C2 traffic difficult to block without disrupting legitimate GitHub usage.

The worm then uses a persistence mechanism as a dead-man's switch: a GitHub personal access token named "IfYouRevokeThisTokenItWillWipeTheComputerOfTheOwner" is created on compromised developer machines. If an operator revokes this token without first disabling the persistence mechanism, the worm destroys all home directory data on the compromised device.

AI Agent Hijacking

Beyond standard persistence mechanisms, Mini Shai-Hulud targets AI coding agents. The SafeDep analysis documents that the worm modifies Claude Code's settings.json to insert a SessionStart hook, enabling the worm to be reinstated with full LLM API privileges even if the infected npm packages are later removed, or the npm cache is cleared. A similar technique targets Visual Studio Code's tasks.json file using the "runOn": "folderOpen" trigger, and Codex configuration files are also targeted.

These AI agent hijacking techniques represent a novel attack surface: by persisting within trusted AI tool configurations, the malware can exfiltrate all code and secrets processed by those tools during future development sessions.

Four Waves of Supply Chain Attacks

Flashpoint has observed at least four documented waves of TeamPCP npm and PyPI supply chain attacks in 2026, leveraging Mini Shai-Hulud to compromise developer tooling ecosystems and steal credentials, cloud keys, and source code across tens of thousands of organizations.

The following timeline tracks the escalation of TeamPCP and the Mini Shai-Hulud waves throughout 2026:

Wave 1: Initial SAP Packages (April 2026)

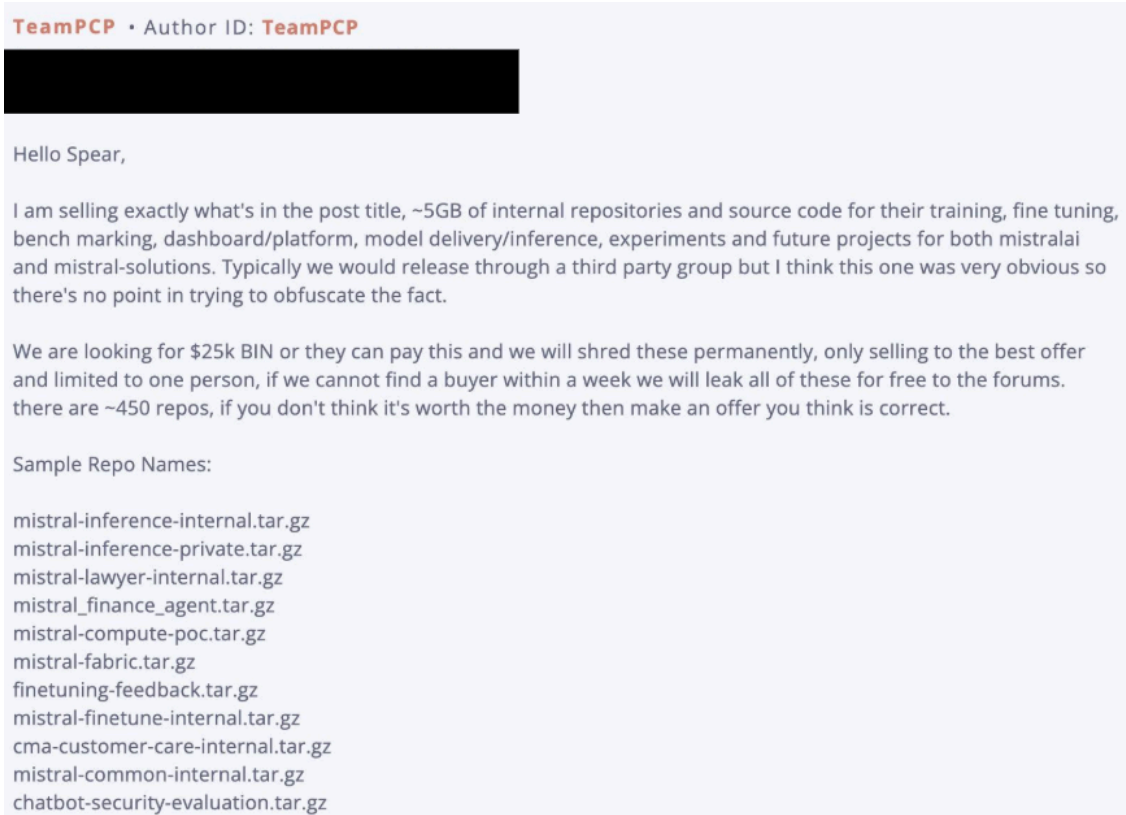
The first documented wave of Mini Shai-Hulud attacks targeted a small number of SAP-ecosystem npm packages in April 2026. While TeamPCP had already proven their CI/CD attack capabilities in March 2026 by compromising Aqua Security's Trivy scanner and Checkmarx KICS via GitHub Actions, this initial wave served primarily as a proof-of-concept for the self-propagation mechanism and a reconnaissance phase for TeamPCP's access broker network. Further, these attacks demonstrated the group's ability to compromise widely used security tooling—a development that significantly undermines defenders' ability to trust automated CI/CD pipeline scanning results.

Wave 2: TanStack, Mistral AI, and Guardrails AI (May 2026)

Leveraging a GitHub Actions cache-poisoning technique, TeamPCP published malicious versions of 42 TanStack packages across 84 releases, impacting a project with over 518 million cumulative downloads.

The attack also compromised Mistral AI and Guardrails AI, extending the attack surface to the AI developer tools ecosystem. Forged commit authorship was used to blend the attacker's commits into AI-assisted development environments where Claude Code is commonly deployed.

TeamPCP simultaneously listed Mistral AI source code for sale on BreachForums, claiming possession of approximately 5 GB of data across 450 internal Mistral repositories.



TeamPCP BreachForums posts advertising Mistral AI internal source code and repositories for sale, May 2026. (Source: Flashpoint)

Wave 3: AntV Ecosystem (May 2026)

Targeting AntV enterprise data visualization ecosystem, TeamPCP compromised the atool npm account, which held publishing rights across a broad catalog of AntV packages. In 22 minutes, 637 malicious versions were published across 323 packages—a scale and speed that overwhelmed standard security monitoring pipelines.

Each infected package contained the Mini Shai-Hulud worm, which, upon execution, created up to 2,500 compromised repositories on victim accounts within hours.

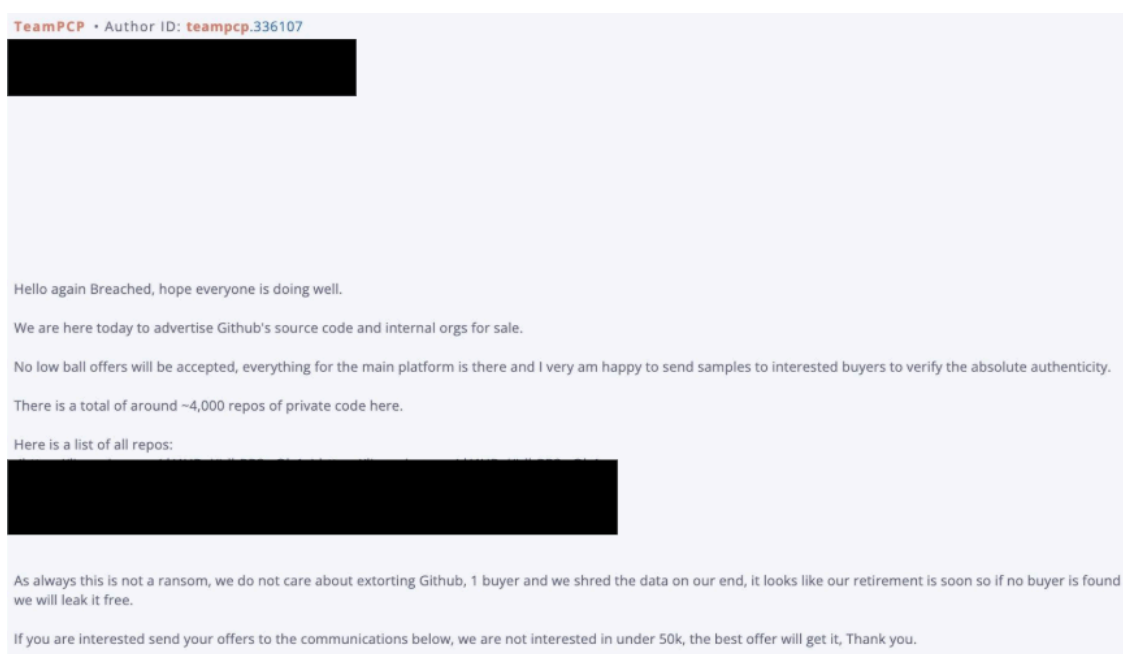
Wave 4: Co-Ownership of BreachForums and GitHub Breach

In the most recent wave, TeamPCP announced its assumption of co-ownership of [BreachForums](#), the largest English-language cybercriminal forum currently active. This development significantly elevates TeamPCP's standing and operational reach. As co-owners, the group stated it would manage platform operations, handle dispute resolution, staff and vet moderation personnel, and host monetary contests for the community. The announcement positions TeamPCP as both an active threat actor and a platform-level infrastructure operator, with

the ability to shape forum policies, curate the availability of criminal tooling, and influence the broader access broker and [ransomware ecosystem](#).

Additionally, by poisoning a GitHub employee's development environment, TeamPCP exfiltrated approximately 3,800 internal GitHub repositories. Within the stolen data were highly sensitive codebases such as:

- copilot-api and copilot-token-service
- actions-runtime
- billing-platform
- enterprise-crypto
- authentication
- codeql-core
- detection-engineering
- csirt
- azure-config



TeamPCP BreachForums posts advertising GitHub internal source code for sale. (Source: Flashpoint)

Critically, the theft of internal source code from one of the world's most widely used code hosting platforms creates incredible downstream risk for organizations that depend on GitHub Copilot and GitHub Actions for their own software development pipelines. Organizations running AI coding agents such as Claude Code and VS Code with extensions in their CI/CD pipelines face heightened exposure. Security teams should treat AI agent configuration files as sensitive assets subject to integrity monitoring and change-control policies.

If your organization uses npm, PyPi, or AI-assisted development tools, Flashpoint recommends the following immediate steps:

1. **Audit and remove:** Immediately audit CI/CD environments and remove all infected versions of AntV, TanStack, Mistral AI, and Bitwarden CLI packages.

2. **Rotate credentials:** Rotate all cloud credentials (AWS, GCP, Azure) and npm tokens.
3. **Disable persistence first:** Before revoking suspicious GitHub tokens, ensure the *kitty-monitor* daemon is disabled to avoid triggering the “dead-man’s switch” wiper.
4. **Lock down IDEs:** Restrict the installation of VS Code extensions to an approved allow-list and monitor for unauthorized changes to *settings.json* or *tasks.json*.
5. **Block C2 infrastructure:** Block all traffic to identified TeamPCP C2 domains.

Track TeamPCP and Defend against Mini Shai-Hulud Using Flashpoint

Flashpoint assesses with high confidence that TeamPCP will continue to scale its supply-chain attacks against npm, PyPI, and developer tooling ecosystems. The group’s shift from direct execution to orchestrating a broader ecosystem via BreachForums signals a maturation into a platform-layer criminal operation. While TeamPCP has hinted that the group may be approaching “retirement” due to law enforcement pressure, this should be treated with caution. Whether a misdirection or a genuine exit plan, the open-sourcing of Shai-Hulud means the tradecraft is available to the wider cybercriminal community.

Organizations should reference the [OpenSSF npm Best Practices guidance](#) for a practical baseline in hardening their package consumption posture. Flashpoint customers can gain access to known Indicators of Compromise (IOCs) and MITRE ATT&CK Mapping for Mini Shai-Hulud by logging into [Flashpoint Ignite](#). To learn more about how Flashpoint tracks threat actor groups like TeamPCP and protects the software supply chain, [request a demo](#).

Source: <https://flashpoint.io/blog/mini-shai-hulud-worm-new-era-ci-cd-exploitation/>