"Shai-Hulud" Worm Compromises npm Ecosystem in Supply Chain Attack (Updated November 26)

By Justin Moore

Published: 2025-11-25 · Archived: 2026-04-29 02:08:33 UTC

Executive Summary

Update: Nov. 25, 2025

Unit 42 researchers investigated a renewed npm-focused compromise, in a campaign dubbed Shai-Hulud 2.0. This was first reported in early November 2025. The current campaign is significantly wider in scope, affecting tens of thousands of GitHub repositories This includes over 25,000 malicious repositories across about 350 unique users.

Notable Differences in November Campaigns

- Execution during pre-install dramatically widened the area of impact
- This campaign introduced a far more aggressive fallback mechanism, which could attempt to destroy a user’s home directory
- New payload files are named setup_bun.js and bun_environment.js
- Stolen credentials and secrets are exfiltrated to public GitHub repositories with the repository description: “Sha1-Hulud: The Second Coming.”

The Shai-Hulud 2.0 campaign represents an aggressive escalation in software supply chain attacks, moving beyond its predecessor's methods by changing the point of infection. By targeting the pre-install phase of software dependencies, the malware achieves two significant breakthroughs:

- It completely eliminates the need for human interaction, guaranteeing execution on virtually every build server processing the infected package
- It effectively bypasses static scanning tools that inspect code during later build stages

While this threat still focuses on stealing high-value cloud credentials, it can also cripple an enterprise's entire CI/CD pipeline. This could disrupt development and potentially lock out internal systems, escalating the attack from simple espionage into a highly disruptive denial-of-service event.

Read the Current Scope of the Attack section for more technical details.

In September, Unit 42 investigated the novel, self-replicating worm as "Shai-Hulud," responsible for the compromise of hundreds of software packages.

This attack represents a significant evolution in supply chain threats, leveraging automated propagation to achieve scale. Unit 42 also assesses with moderate confidence that an LLM was used to generate the malicious bash script,

based on inclusion of comments and emojis.

Palo Alto Networks customers are better protected from, and receive mitigations for aspects of this attack, through various products and services, including:

- Cortex Cloud
- Prisma Cloud
- Advanced URL Filtering
- Advanced WildFire
- Next-Generation Firewall with Advanced Threat Prevention

The Unit 42 Incident Response team can also be engaged to help with a compromise or to provide a proactive assessment to lower your risk.

Related Unit 42 Topics	**Supply Chain**, **Credential Harvesting**, **Phishing**, **JavaScript**

Background on npm Packages and the Supply Chain

The attack may originate from a credential-harvesting phishing campaign spoofing npm and asking developers to "update" their multi-factor authentication (MFA) login options. Once initial access was gained, the threat actor deployed a malicious payload that functions as a worm, initiating a multi-stage attack sequence. Based on the inclusion of comments and emojis in the bash script, Unit 42 assesses with moderate confidence the threat actor leveraged LLM to assist in writing the malicious code.

The malicious package versions contain a worm that executes a post-installation script. This malware scans the compromised environment for sensitive credentials, including:

- .npmrc files (for npm tokens)
- Environment variables and configuration files specifically targeting GitHub Personal Access Tokens (PATs) and API keys for cloud services like:
    - Amazon Web Services (AWS)
    - Google Cloud Platform (GCP)
    - Microsoft Azure

Harvested credentials are exfiltrated to an actor-controlled endpoint. The malware programmatically creates a new public GitHub repository named "Shai-Hulud" under the victim's account and commits the stolen secrets to it, exposing them publicly.

Using the stolen npm token, the malware authenticates to the npm registry as the compromised developer. It then identifies other packages maintained by that developer, injects malicious code into them, and publishes the new, compromised versions to the registry. This automated process allows the malware to spread exponentially without direct actor intervention.

Current Scope of the Attack

As of November 2025, there is a a renewed npm-focused compromise in a campaign dubbed “Shai-Hulud 2.0.”

- **Execution during pre-install (instead of post-install):** Dramatically widened the area of impact across developer machines and continuous integration and continuous delivery (CI/CD) pipelines.
- **A far more aggressive fallback mechanism:** This shifts the tactics from purely data theft to punitive sabotage. If the malware fails to steal credentials, obtain tokens or secure any exfiltration channel (i.e., it cannot authenticate to GitHub, create a repository or find GitHub/npm tokens) it attempts to destroy the victim’s entire home directory. It does so by securely overwriting and deleting every writable file owned by the current user under their home folder.
- **New payload files:** These are named setup_bun.js and bun_environment.js. The attack disguises itself as a helpful Bun installer. The core payload, bun_environment.js, is a massive file (over 10 MB) that uses extreme obfuscation techniques. It delays full execution on developer machines by forking itself into a detached background process. This allows the original install process to exit cleanly, giving the user the illusion of a normal installation.
- **Sha1-Hulud:** Stolen credentials and secrets are exfiltrated to public GitHub repositories with the repository description: “Sha1-Hulud: The Second Coming.” It also attempts persistence by creating a GitHub Actions workflow file named discussion.yaml. This workflow registers the infected machine as a self-hosted runner and allows attackers to execute arbitrary commands by opening GitHub discussions.

Scope of the Attack Before November 2025

The scope of the compromise is extensive, impacting numerous packages, including the widely used @ctrl/tinycolor library, which receives millions of weekly downloads.

Credential theft from this campaign can lead directly to compromise of cloud services (such as AWS, Azure, GCP), leading to data theft from storage buckets, ransomware deployment, cryptomining or deletion of production environments. It may also lead to direct database theft and hijacking of third-party services for phishing. Additionally, stolen SSH keys can enable lateral movement within compromised networks.

Interim Guidance

1. Credential Rotation: Immediately rotate all developer credentials. This includes npm access tokens, GitHub PATs and SSH keys, and all programmatic access keys for cloud and third-party services. Assume that any secret present on a developer's machine may have been compromised.
2. Dependency Auditing: Conduct a thorough and immediate audit of all project dependencies. Use tools like npm audit to identify vulnerable package versions. Scrutinize your project's package-lock.json or yarn.lock files to ensure you are not using any of the known-compromised packages. Remove or update affected dependencies immediately.
3. GitHub Account Security Review: All developers should review their GitHub accounts for unrecognized public repositories (specifically "Shai-Hulud"), suspicious commits or unexpected modifications to GitHub Actions workflows that could establish persistence.
4. Enforce MFA: Ensure that MFA is strictly enforced on all developer accounts, particularly for critical platforms like GitHub and npm, to prevent credential abuse.

Unit 42 Managed Threat Hunting Queries

```
// Description: Check for connections to any webhook.site domains in raw NGFW URL logs. Optional filter for specific URI observed in use by threat actor.
dataset = panw_ngfw_url_raw
| filter lowercase(url_domain) contains "webhook.site"
| alter susp_uri = if(uri contains "bb8ca5f6-4175-45d2-b042-fc9ebb8170b7")
// Optional filter:
// | filter susp_uri = true
| fields url_domain, uri, susp_uri, *
```

```
// Description: Check for connections to any webhook.site domains in XDR telemetry. Optional filter for specific URI observed in use by threat actor.
dataset = xdr_data
| filter event_type = STORY
| filter lowercase(dst_action_external_hostname) contains "webhook.site" or lowercase(dns_query_name) contains "webhook.site"
| fields agent_hostname, dst_action_external_hostname, dns_query_name
```

```
// Description: Detect malicious YAML file
```

```
dataset = xdr_data
| filter event_type = FILE and action_file_name = "shai-hulud-workflow.yml" and agent_os_type in (ENUM.AGENT_OS_MAC, ENUM.AGENT_OS_LINUX)
| fields agent_hostname, actor_effective_username, action_file_name, action_file_path, actor_process_image_name, actor_process_command_line
```

```
// Description: Detects Trufflehog usage. Legitimate tool abused by threat actor for secrets discovery. False positives may occur if there is legitimate use.
dataset = xdr_data
| filter event_type = PROCESS and lowercase(action_process_image_command_line) contains "trufflehog"
| fields agent_hostname, actor_effective_username, actor_process_command_line, action_process_image_command_line
```

Updated Queries for November 2025 Campaign

```
// Description: Detect malicious bundle.js, bun_environment.js, and setup_bun.js files
preset = xdr_file
| fields agent_hostname, action_file_name, action_file_path, event_type, event_sub_type, actor_process_image_name, actor_process_command_line, action_file_sha256
| filter event_type = ENUM.FILE
| filter action_file_sha256 = "46faab8ab153fae6e80e7cca38eab363075bb524edd79e42269217a083628f09" // bundle.js from September 2025 attack
    or action_file_sha256 in ("62ee164b9b306250c1172583f138c9614139264f889fa99614903c12755468d0", "f099c5d9ec417d4445a0328ac0ada9cde79fc37410914103ae9c609cbc0ee068", "cbb9bc5a8496243e02f3cc080efbe3e4a1430ba0671f2e43a202bf45b05479cd") // bun_environment.js from November 2025 attack
```

```
    or action_file_sha256 =
"a3894003ad1d293ba96d77881ccd2071446dc3f65f434669b49b3da92421901a" // setup_bun.js from
November 2025 attack
```

```
// Description: Detects the unique SHA1HULUD string used in runner creation
preset = xdr_process
| fields agent_hostname, actor_effective_username, action_process_image_name,
action_process_image_path, action_process_image_command_line, actor_process_image_name,
actor_process_image_path, actor_process_command_line, agent_os_type, event_type, event_sub_type
| filter event_type = ENUM.PROCESS
    and event_sub_type = ENUM.PROCESS_START
| filter action_process_image_command_line contains " --name SHA1HULUD"
```

```
// Description: Detects an extremely large (>=9MB) bun_environment.js file. False positives are possible,
be sure to check action_file_path for the package name and version of any hits.
preset = xdr_file
| fields agent_hostname, action_file_name, action_file_path, action_file_size, event_type,
event_sub_type, actor_process_image_name, actor_process_command_line, action_file_sha256
| filter event_type = ENUM.FILE
    and event_sub_type = ENUM.FILE_WRITE
| filter action_file_name = "bun_environment.js"
    and action_file_size >= 9437184
```

Conclusion

The Shai-Hulud worm represents a significant escalation in the ongoing series of npm attacks targeting the open-source community. This follows recent incidents such as the s1ngularity/Nx compromise, which involved credential theft and exposed private repositories, and a widespread npm phishing campaign observed in September 2024.

Its self-replicating design is particularly notable, effectively combining credential harvesting with an automated dissemination mechanism that exploits maintainers' existing publishing rights to proliferate across the ecosystem. Furthermore, we have observed the integration of AI-generated content within the Shai-Hulud campaign, a

development that follows the s1ngularity/Nx attack's explicit weaponization of AI command-line tools for reconnaissance. This signifies the ever-evolving threat from malicious actors exploiting AI for malicious activity, accelerating secret sprawl.

The consistent and refined nature of these attack methodologies underscores a growing threat to open-source software supply chains. These attacks are propagating at the speed of Continuous Integration and Continuous Delivery (CI/CD), which poses long-lasting and increasing security challenges for the entire ecosystem.

Palo Alto Networks has shared our findings with our fellow Cyber Threat Alliance (CTA) members. CTA members use this intelligence to rapidly deploy protections to their customers and to systematically disrupt malicious cyber actors. Learn more about the Cyber Threat Alliance.

Palo Alto Networks Product Protections and Detections for npm Packages Supply Chain Attacks

Palo Alto Networks customers can leverage a variety of product protections, services and updates designed to identify and defend against this threat.

If you think you might have been compromised or have an urgent matter, get in touch with the Unit 42 Incident Response team or call:

- North America: Toll Free: +1 (866) 486-4842 (866.4.UNIT42)
- UK: +44.20.3743.3660
- Europe and Middle East: +31.20.299.3130
- Asia: +65.6983.8730
- Japan: +81.50.1790.0200
- Australia: +61.2.4062.7950
- India: 000 800 050 45107

Advanced WildFire

The Advanced WildFire machine-learning models and analysis techniques have been reviewed and updated in light of indicators associated with this threat.

Next-Generation Firewalls With Advanced Threat Prevention

Next-Generation Firewall with the Advanced Threat Prevention security subscription can help block the attack via the following Threat Prevention signatures 87042, 87046 and 87047.

Cloud-Delivered Security Services for the Next-Generation Firewall

Advanced URL Filtering helps to block meddler-in-the-middle (MitM) phishing attacks and classifies as malicious URLs associated with this activity.

Cortex XDR and XSIAM

Cortex XDR and XSIAM agents help protect against the threats described in this article. The agents prevent the execution of known malware and may also prevent the execution of unknown malware using Behavioral Threat Protection and machine learning based on the Local Analysis module.

Cortex Cloud

Cortex Cloud offers extensive ASPM and supply chain security capabilities to help identify the vulnerabilities and misconfigurations that Shai-Hulud exploits. With real-time SBOM visibility, teams can instantly query their inventory against known malicious npm packages. The platform's Operational Risk model adds another layer of defense by evaluating open-source components based on maintainer activity, deprecation signals, and community health to flag risky packages even without published CVEs.

To harden pipelines, Cortex Cloud provides out-of-the-box CI/CD rules aligned with OWASP and CIS guidance, including checks for missing npm lock files, insecure “npm install” usage, git-sourced packages without commit hashes, and unused dependencies that expand the attack surface.

Since CVE publication often lags behind active attacks it’s critical to review and verify that your applications are not relying on unsanctioned npm package versions. Together, these controls help ensure malicious versions can’t silently enter builds or linger in your environment.

Cortex Cloud has published a detailed blog post describing how Cortex Cloud can be used for detecting and preventing supply chain attacks.

Prisma Cloud

Prisma Cloud can help detect the use of the malicious packages and recognize misconfigurations in the pipelines that might lead customers to use untested/unsanctioned OSS package versions. However, the scanner is designed for detection of vulnerabilities, license issues and operational risks, and not for detecting malicious code on new packages. It is important to investigate relevant CI/CD alerts and ensure your applications are not using unsanctioned versions of npm packages.

Indicators of Compromise

- 46faab8ab153fae6e80e7cca38eab363075bb524edd79e42269217a083628f09
- b74caeaa75e077c99f7d44f46daaf9796a3be43ecf24f2a1fd381844669da777
- dc67467a39b70d1cd4c1f7f7a459b35058163592f4a9e8fb4dffcbba98ef210c
- 4b2399646573bb737c4969563303d8ee2e9ddbd1b271f1ca9e35ea78062538db
- hxxps://webhook[.]site/bb8ca5f6-4175-45d2-b042-fc9ebb8170b7

Additional Resources

- Breakdown: Widespread npm Supply Chain Attack Puts Billions of Weekly Downloads at Risk – Palo Alto Networks
- Sha1-Hulud: The Second Coming - Zapier, ENS Domains, and Other Prominent NPM Packages Compromised – StepSecurity

- Shai-Hulud 2.0 Supply Chain Attack: 25K+ Repos Exposing Secrets – Blog, Wiz

Updated Sept. 18, 2025 at 2:25 p.m. PT, to add product protections for Advanced Threat Prevention and update protections for Cortex Cloud

Updated Sept. 19, 2025 at 3:50 p.m. PT, to add product protections for Advanced URL Filtering and update protections for Cortex Cloud

Updated Sept. 23, 2025 at 4:36 p.m. PT, to add additional Threat Prevention signatures

Updated Nov. 25, 2025 at 8:00 a.m. PT, to update Executive Summary and Scope of Attack sections to include information on second campaign

Updated Nov. 26, 2025 at 8:10 a.m. PT, to update Managed Threat Hunting queries and Cortex Cloud protection information

Updated Dec. 3, 2025 at 5:45 a.m. PT, to update Cortex product protection information

Table of Contents

-
- Executive Summary
  - Update: Nov. 25, 2025
    - Notable Differences in November Campaigns
- Background on npm Packages and the Supply Chain
- Current Scope of the Attack
- Scope of the Attack Before November 2025
- Interim Guidance
- Unit 42 Managed Threat Hunting Queries
  - Updated Queries for November 2025 Campaign
- Conclusion
- Palo Alto Networks Product Protections and Detections for npm Packages Supply Chain Attacks
  - Advanced WildFire
  - Next-Generation Firewalls With Advanced Threat Prevention
  - Cloud-Delivered Security Services for the Next-Generation Firewall
  - Cortex XDR and XSIAM
  - Cortex Cloud
  - Prisma Cloud
- Indicators of Compromise
- Additional Resources

Related Articles

- The npm Threat Landscape: Attack Surface and Mitigations
- Threat Brief: Escalation of Cyber Risk Related to Iran (Updated April 17)
- Threat Brief: Widespread Impact of the Axios Supply Chain Attack

Source: https://unit42.paloaltonetworks.com/npm-supply-chain-attack/