Mitigating Risks from the Shai-Hulud NPM Worm | ThreatLabz

By Atinderpal Singh

Published: 2025-09-19 · Archived: 2026-04-06 01:34:34 UTC

Zscaler Blog

Get the latest Zscaler blog updates in your inbox

Introduction

On September 15th 2025, ReversingLabs (RL) researchers discovered a self-replicating worm called “Shai-Hulud” in the `npm` open-source registry. The worm autonomously spreads through the `npm` registry by hijacking maintainer accounts and injecting malicious code into public and private packages. Over 200 `npm` packages and more than 500 versions were compromised between September 14th and 18th. Each infected package helps the Shai-Hulud worm spread further which creates a chain reaction across the `npm` ecosystem.

Named after its repository, the Shai-Hulud worm targets sensitive data like tokens, keys, and private repositories. While end-user applications are less directly affected, build environments may have been exposed through leaked credentials or code. RL has identified hundreds of compromised packages, including widely used ones like ngx-bootstrap, ng2-file-upload, and @ctrl/tinycolor, which have millions of weekly downloads. The interconnected nature of `npm` packages makes it difficult to predict the worm’s impact.

Recommendations

- Use private registry proxies and software composition analysis (SCA) tools to filter and monitor third-party packages. Remove compromised package versions, clear caches, and reinstall clean ones. Use private package managers to block malicious versions.
- Apply least privilege principles by using scoped, short-lived keys and tokens. Revoke npm tokens, GitHub personal access tokens (PATs), cloud keys, and CI/CD secrets.
- Flag abnormal `npm` publish events, GitHub workflow additions, or the unexpected use of secret scanners in CI processes. Hunt for indicators of compromise (IOCs) like *bundle.js*, workflows named *shai-hulud-workflow.yml*, or outbound traffic to *webhook[.]site*.
- Update response playbooks for supply chain attacks and conduct practice drills. Treat impacted systems as compromised by isolating, scanning, or reimaging them.
- Restrict build environments to internal package managers or trusted mirrors, and limit internet access to reduce data exfiltration risks. Enable multifactor authentication (MFA) across all platforms, including `npm`, GitHub, and cloud services.
- Reinforce phishing awareness, and the secure handling of tokens and secrets with developer teams.

Affected Versions

Notable examples of compromised packages and their versions include:

- **@ctrl/tinycolor** - Versions 4.1.1 and 4.1.2
- **@crowdstrike/*** - Multiple versions of packages

Impacted platforms

All major operating systems (OS), including Windows, Linux, and macOS, are affected and become vulnerable when compromised `npm` packages are installed.

Vulnerability Details

The Shai-Hulud worm exploits compromised `npm` packages by planting a malicious post-install script. When executed, the script executes several actions:

- Uses TruffleHog to steal sensitive data, such as tokens, API keys, environment variables, and cloud credentials.
- Sends exfiltrated data to threat actor-controlled webhooks and GitHub repositories named *Shai-Hulud.*
- Publishes infected versions of all packages owned by the victim.
- Injects malicious workflows and converts private repositories to public access.

This combination of **credential theft**, **package trojanization**, and **self-replication** makes the Shai-Hulud worm uniquely dangerous.

Conclusion

The Shai-Hulud worm rapidly compromised hundreds of `npm` packages and versions across Windows, Linux, and macOS, showing how quickly and widely vulnerabilities in open-source ecosystems can be exploited. By combining credential theft, automated propagation, and repository tampering, the Shai-Hulud worm has set a precedent for future supply chain attacks. To prevent similar incidents, organizations must act immediately by revoking exposed credentials, strengthening supply chain defenses, and implementing enhanced monitoring to detect and respond to potential threats.

Zscaler Coverage

Zscaler has enhanced its security measures to cover this threat, ensuring that any attempts to download a malicious `npm` package will be detected under the following threat classifications:

Advanced Threat Protection

- JS/Shulud.A
- JS.Malicious.npmpackage

Attempts to access the web service for data exfiltration will be identified and flagged under the following threat name:

Advanced Threat Protection

- JS.Worm.Shai-Hulud.LZ

Thank you for reading

Get the latest Zscaler blog updates in your inbox
