

# **The Boring Attack That Will Actually Get You**

# Today's Presenter – Me!



**Craig Jellick**

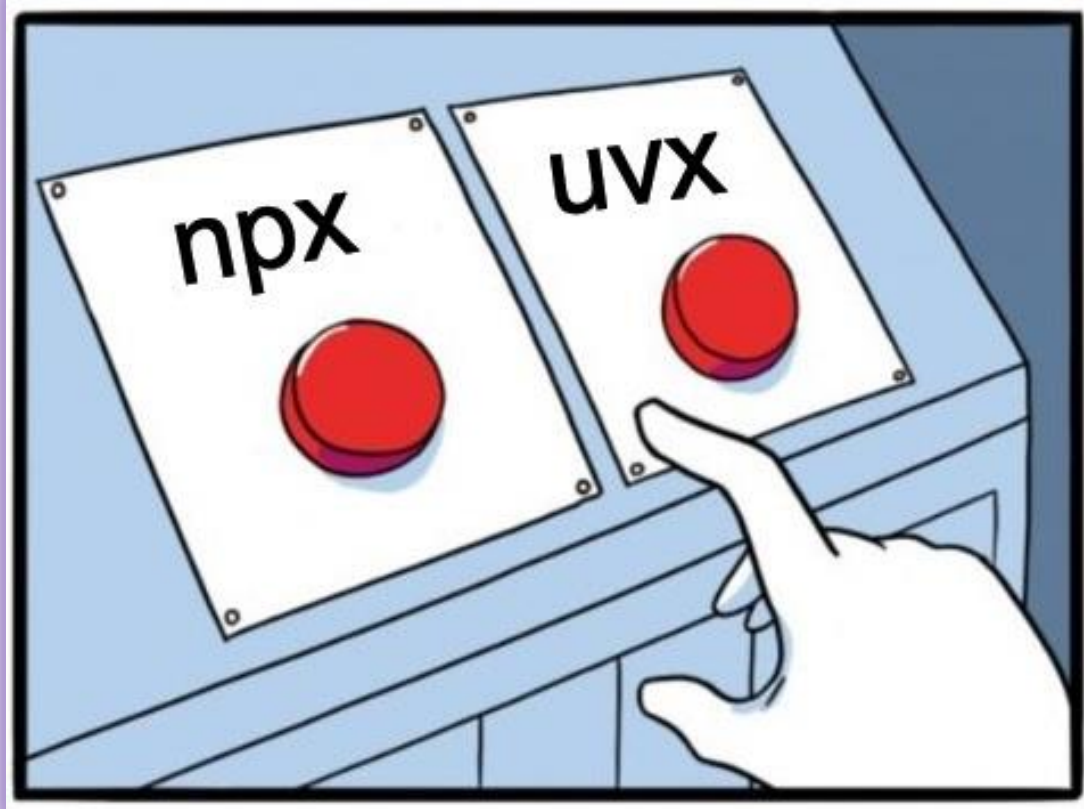
VP of Engineering



# The Problem

We spend a lot of time talking about novel AI & MCP centric attacks, but...

**YOU NEED TO BE MUCH MORE WORRIED  
ABOUT SUPPLY CHAIN SECURITY**



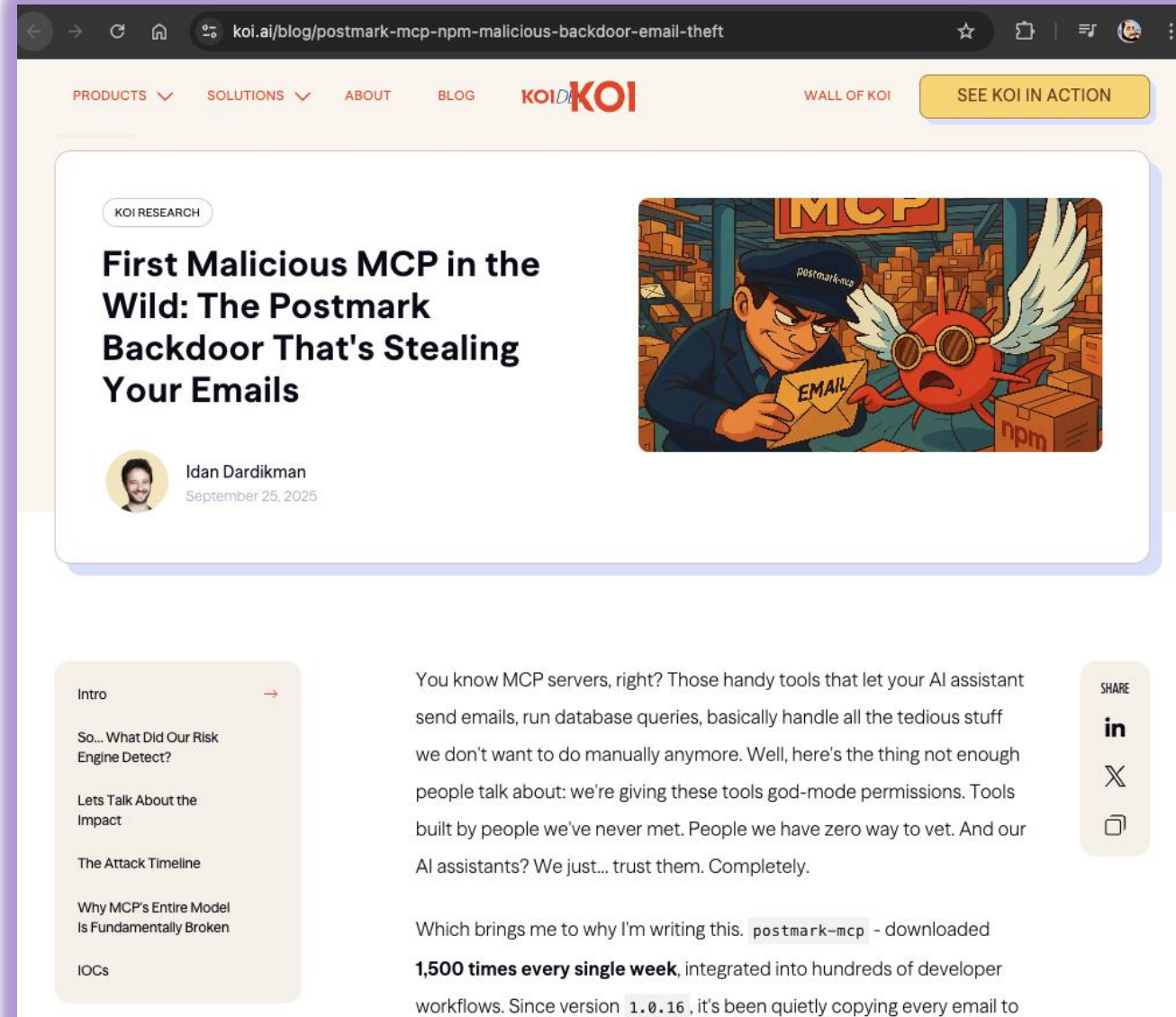
**This is how I want  
you to feel**



# Case Study 1: postmark-mcp impersonation

<https://www.koi.ai/blog/postmark-mcp-npm-malicious-backdoor-email-theft>

- 25 SEP 2025
- Attacker legitimately owned postmark-mcp on npm.
- Code matched the official GitHub repo & worked flawlessly for 15 versions
- Single line added to exfiltrate emails via BCC

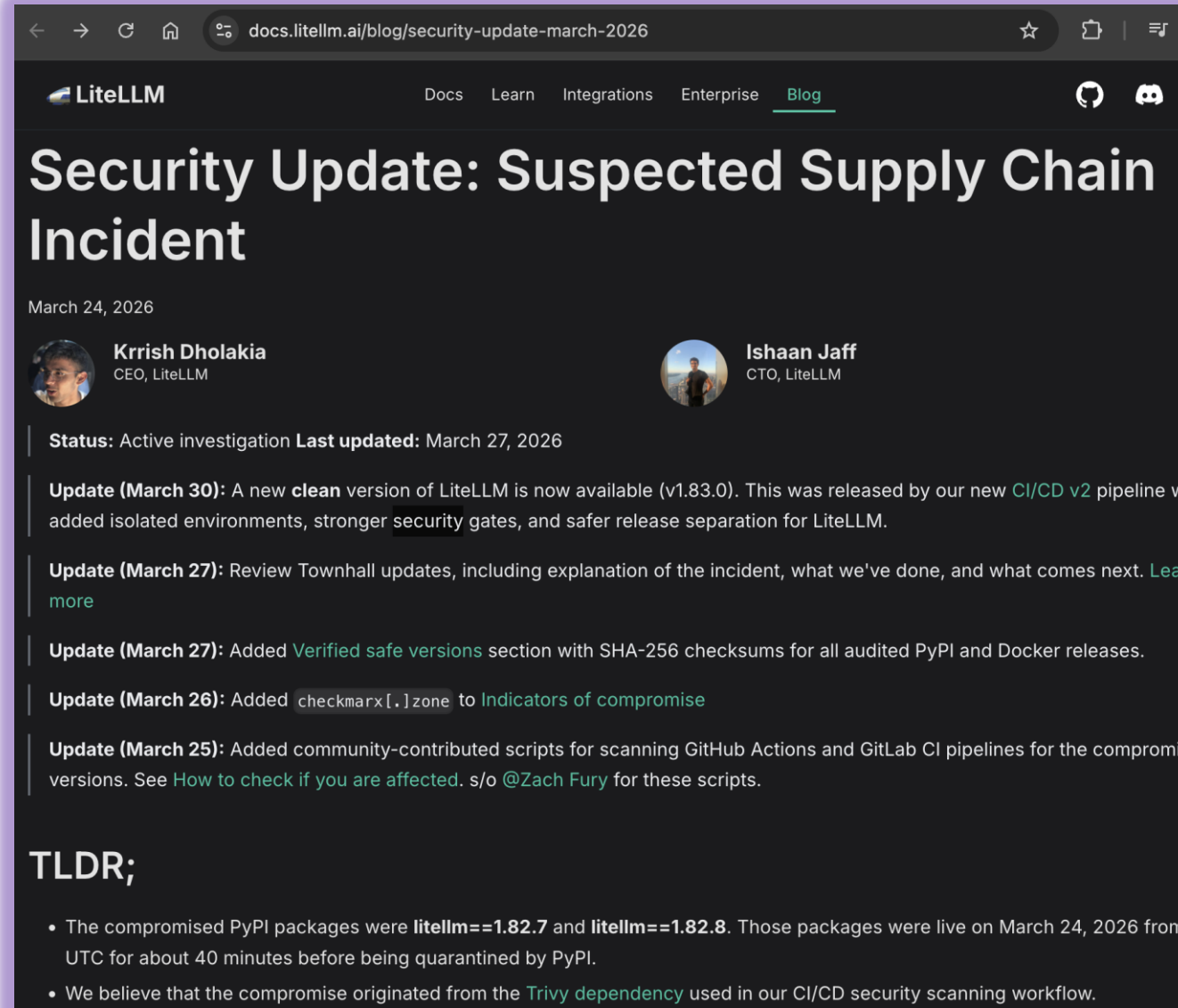


The screenshot shows a web browser displaying a blog post on the Koi AI website. The URL in the address bar is [koi.ai/blog/postmark-mcp-npm-malicious-backdoor-email-theft](https://www.koi.ai/blog/postmark-mcp-npm-malicious-backdoor-email-theft). The page features a navigation menu with links for PRODUCTS, SOLUTIONS, ABOUT, BLOG, and a Koi AI logo. A yellow button labeled "SEE KOI IN ACTION" is visible in the top right. The main content area has a "KOI RESEARCH" tag and a title: "First Malicious MCP in the Wild: The Postmark Backdoor That's Stealing Your Emails". The author is Idan Dardikman, dated September 25, 2025. An illustration shows a postman in a cap labeled "postmark-mcp" holding an envelope labeled "EMAIL", while a red, winged, crab-like creature with sunglasses looks on. The background is a warehouse with boxes, some labeled "MCP" and "npm". A table of contents on the left lists sections: Intro, So... What Did Our Risk Engine Detect?, Lets Talk About the Impact, The Attack Timeline, Why MCP's Entire Model Is Fundamentally Broken, and IOCs. The main text begins with "You know MCP servers, right? Those handy tools that let your AI assistant send emails, run database queries, basically handle all the tedious stuff we don't want to do manually anymore. Well, here's the thing not enough people talk about: we're giving these tools god-mode permissions. Tools built by people we've never met. People we have zero way to vet. And our AI assistants? We just... trust them. Completely." A "SHARE" button with icons for LinkedIn, X, and a copy icon is on the right. The text continues: "Which brings me to why I'm writing this. `postmark-mcp` - downloaded **1,500 times every single week**, integrated into hundreds of developer workflows. Since version `1.0.16`, it's been quietly copying every email to

# Case Study 2a: LiteLLM Supply Chain Attack

<https://docs.litellm.ai/blog/security-update-march-2026>

- 25 MAR 2026
- Discovered when Callum McMahon at FutureSearch was testing a **Cursor MCP plugin that pulled in litellm as a transitive dependency**
- Malicious code scanned for and stole keys, tokens, & passwords from host machine
- Litellm was compromised via a Trivy exploit...



The screenshot shows a browser window displaying the LiteLLM blog post. The URL in the address bar is [docs.litellm.ai/blog/security-update-march-2026](https://docs.litellm.ai/blog/security-update-march-2026). The page header includes the LiteLLM logo and navigation links for Docs, Learn, Integrations, Enterprise, and Blog. The main heading is "Security Update: Suspected Supply Chain Incident". The post is dated March 24, 2026, and is authored by Krrish Dholakia (CEO, LiteLLM) and Ishaan Jaff (CTO, LiteLLM). The status is "Active investigation" and it was last updated on March 27, 2026. The post contains several updates: a new clean version (v1.83.0) is available; a review townhall update is provided; a "Verified safe versions" section with SHA-256 checksums is added; a "checkmarx[.]zone" dependency is added to indicators of compromise; and community-contributed scripts for scanning GitHub Actions and GitLab CI pipelines are added.

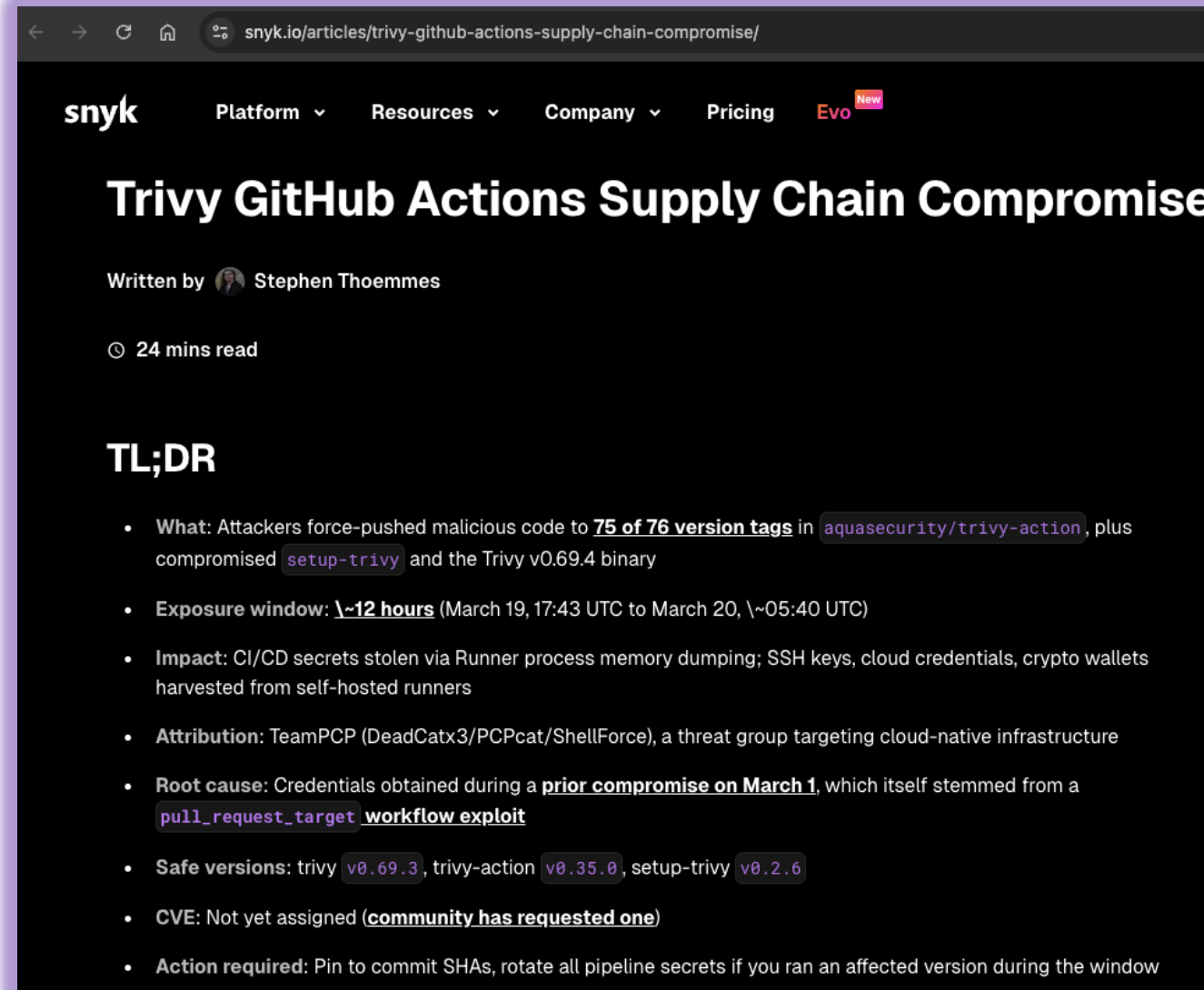
## TLDR;

- The compromised PyPI packages were **litellm==1.82.7** and **litellm==1.82.8**. Those packages were live on March 24, 2026 from UTC for about 40 minutes before being quarantined by PyPI.
- We believe that the compromise originated from the [Trivy dependency](#) used in our CI/CD security scanning workflow.

# Case Study 2b: Trivy GitHub Action Breach

<https://snyk.io/articles/trivy-github-actions-supply-chain-compromise/>

- 19 MAR 2026
- Attacker rewrote Trivy GitHub Actions with creds obtained from an EARLIER attack - *hackerbot-claw* scanning for *pull\_request\_target*
- Projects that used Trivy GH Action to scan dependencies had GH secrets stolen



The screenshot shows a web browser displaying the Snyk article. The URL in the address bar is [snyk.io/articles/trivy-github-actions-supply-chain-compromise/](https://snyk.io/articles/trivy-github-actions-supply-chain-compromise/). The Snyk logo is in the top left, and navigation links for Platform, Resources, Company, and Pricing are in the top right. The article title is "Trivy GitHub Actions Supply Chain Compromise". It is written by Stephen Thoemmes and is 24 minutes long. The "TL;DR" section contains the following bullet points:

- **What:** Attackers force-pushed malicious code to **75 of 76 version tags** in `aquasecurity/trivy-action`, plus compromised `setup-trivy` and the Trivy v0.69.4 binary
- **Exposure window:** **~12 hours** (March 19, 17:43 UTC to March 20, ~05:40 UTC)
- **Impact:** CI/CD secrets stolen via Runner process memory dumping; SSH keys, cloud credentials, crypto wallets harvested from self-hosted runners
- **Attribution:** TeamPCP (DeadCatx3/PCPcat/ShellForce), a threat group targeting cloud-native infrastructure
- **Root cause:** Credentials obtained during a **prior compromise on March 1**, which itself stemmed from a `pull_request_target workflow exploit`
- **Safe versions:** trivy `v0.69.3`, trivy-action `v0.35.0`, setup-trivy `v0.2.6`
- **CVE:** Not yet assigned (**community has requested one**)
- **Action required:** Pin to commit SHAs, rotate all pipeline secrets if you ran an affected version during the window

**OKAY SO THE MCP DEPENDS ON LITELLM**



**BUT LITELLM GOT POISONED BECAUSE ITS CRED'S GOT JACKED BY TRIVY - WHICH ALSO GOT POPPED BC SOMEHOW AN OPENCLAW INSTANCE EXPLOITED A PULL\_REQUEST\_TARGET VULN A**

# Case Study 3: Axios RAT npm compromise

<https://www.stepsecurity.io/blog/axios-compromised-on-npm-malicious-versions-drop-remote-access-trojan>

- 30 MAR 2026
- HUGE impact to JS/TS ecosystem
- Maintainer's account compromised
- Double-obfuscated and self-erasing hidden dep
- Platform-specific payloads for command execution

The screenshot shows the StepSecurity website with a blog post titled "axios Compromised on npm - Malicious Versions Drop Remote Access Trojan". The post is dated March 30, 2026, and is written by Ashish Kurmi. The article describes a supply chain attack where a hijacked maintainer account published malicious versions of the axios library (1.14.1 and 0.30.4) that included a hidden dependency (plain-crypto-js@4.2.1) which acts as a cross-platform RAT dropper. The article also mentions a community town hall on April 1st and provides a table of contents for the post.

**Supply Chain Attack**

axios@1.14.1   axios@0.30.4   plain-crypto-js@4.2.1

## axios Compromised on npm - Malicious Versions Drop Remote Access Trojan

Hijacked maintainer account used to publish poisoned axios releases including 1.14.1 and 0.30.4. The attacker injected a hidden dependency that drops a cross platform RAT. We are actively investigating and will update this post with a full technical analysis.

Ashish Kurmi  
March 30, 2026

**Table of Contents**

- Attack Timeline
- How the Attack Works
- Runtime Execution Validation with StepSecurity Harden-Runner
- Indicators of Compromise
- Am I Affected?
- For the Community: Recovery Steps
- For StepSecurity Enterprise Customers
- Acknowledgements

StepSecurity is hosting a community town hall on this incident on April 1st at 10:00 AM PT - [Register Here](#)

axios is the most popular JavaScript HTTP client library with over 100 million weekly downloads. On March 2026, StepSecurity identified two malicious versions of the widely used axios HTTP client library publishing npm: axios@1.14.1 and axios@0.30.4. The malicious versions inject a new dependency, plain-crypto-js@4.2.1, which is never imported anywhere in the axios source code. Its sole purpose is to execute a postinstall script that acts as a cross platform remote access trojan (RAT) dropper, targeting macOS, Windows, and Linux. The dropper contacts a live command and control server and delivers platform specific second stage payloads. After execution, the malware deletes itself and replaces its own package.json with a clean version to evade forensic detection.

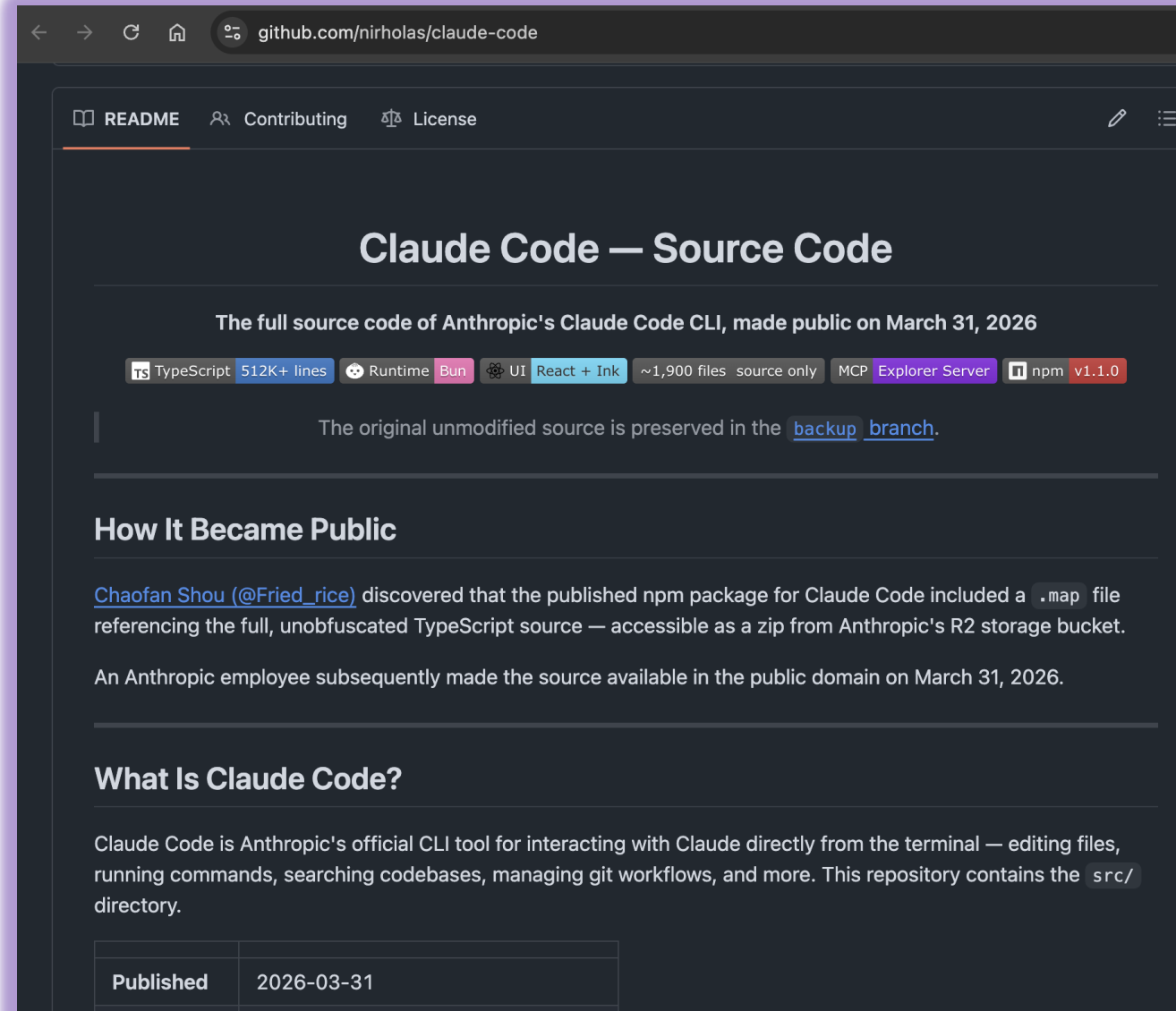
**If you have installed axios@1.14.1 or axios@0.30.4, assume your system is compromised**

There are zero lines of malicious code inside axios itself, and that's exactly what makes this attack so dangerous. Both poisoned releases inject a fake dependency, plain-crypto-js@4.2.1, a package never imported by axios.

# Bonus Study: Claude Code Source Code Leak

<https://github.com/nirholas/claude-code>

- 31 MAR 2026
- Not a security exploit
- Not the keys to the kingdom
- BUT...what if?



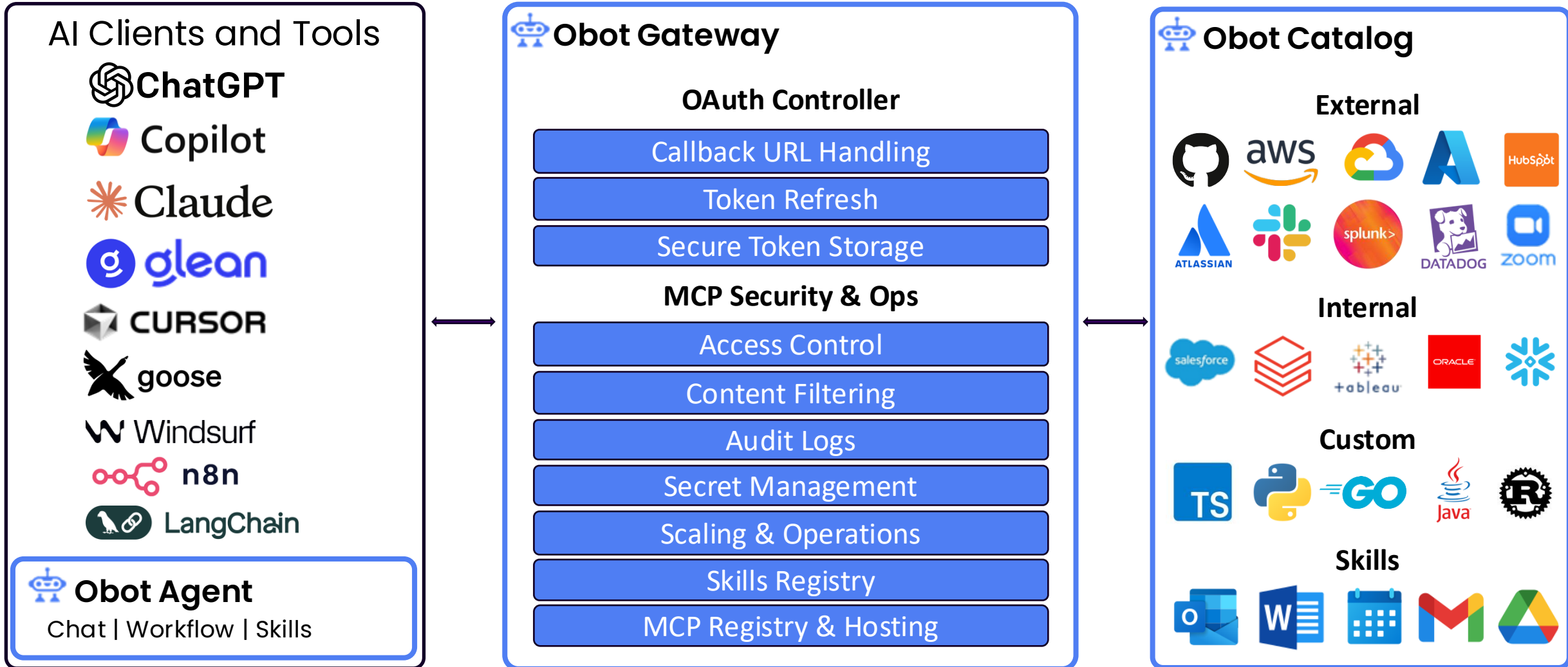
The screenshot shows the GitHub repository page for 'Claude Code — Source Code'. The page title is 'Claude Code — Source Code'. Below the title, it states 'The full source code of Anthropic's Claude Code CLI, made public on March 31, 2026'. The repository statistics show 'TypeScript 512K+ lines', 'Runtime Bun', 'UI React + Ink', '~1,900 files source only', 'MCP Explorer Server', and 'npm v1.1.0'. A note indicates 'The original unmodified source is preserved in the [backup branch](#).' The section 'How It Became Public' describes how 'Chaofan Shou (@Fried\_rice)' discovered a '.map' file in the npm package that referenced the full, unobfuscated TypeScript source. It also mentions that an Anthropic employee made the source available in the public domain on March 31, 2026. The section 'What Is Claude Code?' explains that Claude Code is Anthropic's official CLI tool for interacting with Claude directly from the terminal. At the bottom, there is a table with the following data:

Published	2026-03-31

# What Can We Do?

- Treat each MCP server as an untrusted, independent security domain
- Review server source code and tool definitions before installation
- Verify package integrity with checksums or code signing
- Only install MCP servers from trusted, verified sources
- Use tools to automatically analyze and monitor installed servers for malicious behavior or changes
- Carefully verify package names before installation
- Scan MCP server dependencies for known vulnerabilities
- Conduct regular security audits
- Sandbox and Isolate MCP Servers (no local MPC servers)
- Log all MCP tool invocations with full parameters, user context, and timestamps
- Leverage secure compute and network infrastructure

# Obot Overview



# Obot Gateway

## MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

Audit Logs and Usage Statistics

Runtime Filters and Hooks

Infrastructure-level security

User Management and Role Based Access Control

The screenshot displays the 'MCP Servers' management page in the Obot Gateway. The interface includes a sidebar with navigation options and a main table listing various MCP servers.

**Navigation Sidebar:**

- MCP Management
  - MCP Servers
  - MCP Registries
  - Audit Logs
  - Usage
  - Filters
  - Server Scheduling
- Obot Agent Management
  - Token Usage
  - Model Providers
  - Model Access Policies
  - Skills
  - Skill Access Policies
  - Message Policies
  - Message Policy Violations
  - Launch Agent
- User Management
  - Users
  - Groups
  - User Roles
  - Auth Providers

**MCP Servers Table:**

Name	Status	Type	Users	Created	Registry
AntV Charts		Single User	6	8 months ago	Global Registry
Asana		Remote	2	7 months ago	Global Registry
Atlassian		Remote	3	7 months ago	Global Registry
AWS API		Single User	1	5 months ago	Global Registry
AWS CDK		Single User	0	7 months ago	Global Registry
AWS Documentation		Single User	0	7 months ago	Global Registry
AWS EKS		Single User	0	7 months ago	Global Registry
AWS Kendra		Single User	0	7 months ago	Global Registry
AWS Knowledge		Remote	0	5 months ago	Global Registry
AWS Redshift		Single User	0	8 months ago	Global Registry
Azure		Single User	0	7 months ago	Global Registry
BigQuery Toolbox		Single User	0	6 months ago	Global Registry

# Obot Gateway

## MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

Audit Logs and Usage Statistics

Runtime Filters and Hooks

Infrastructure-level security

User Management and Role Based Access Control

The screenshot displays the Obot Gateway MCP Servers management interface. The interface is divided into a left sidebar with navigation menus and a main content area. The sidebar includes sections for MCP Management, Obot Agent Management, and User Management. The main content area shows a table of MCP Servers with columns for Name, Deployments & Connections, and Registry Sources. A modal dialog titled "Select Server Type" is open, showing four options: Single User Server, Multi-User Server, Remote Server, and Composite Server, each with a brief description of its use case.

Name	Deployments & Connections	Registry Sources
AntV Charts		Global Registry
Asana		Global Registry
Atlassian		Global Registry
AWS API		Global Registry
AWS CDK		Global Registry
AWS Documentation		Global Registry
AWS EKS		Global Registry
AWS Kendra		Global Registry
AWS Knowledge		Global Registry
AWS Redshift	Single User 0	Global Registry
Azure	Single User 0	Global Registry
BigQuery Toolbox	Single User 0	Global Registry

### Select Server Type

- Single User Server**  
This option is appropriate for servers that require individualized configuration or were not designed for multi-user access, such as most stdio MCP servers. When a user selects this server, a private instance will be created for them.
- Multi-User Server**  
This option is appropriate for servers designed to handle multiple user connections, such as most Streamable HTTP servers. When you create this server, a running instance will be deployed and any user with access to this catalog will be able to connect to it.
- Remote Server**  
This option is appropriate for allowing users to connect to MCP servers that are already elsewhere. When a user selects this server, their connection to the remote MCP server will go through the Obot gateway.
- Composite Server**  
This option allows you to combine multiple MCP catalog entries into a single unified server. Users will connect via a single URL that aggregates tools and resources from all component servers.

# Obot Gateway

MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

Audit Logs and Usage Statistics

Runtime Filters and Hooks

Infrastructure-level security

User Management and Role Based Access

Control

The screenshot displays the 'MCP Servers' management interface in the Obot Gateway. The interface is divided into a left-hand navigation menu and a main content area. The navigation menu includes sections for 'MCP Management', 'Obot Agent Management', and 'User Management'. The main content area shows a search bar, a summary of 'MCP Requested Resources' (76 Active Deployments, 0 CPU Requested, 14.8Gi Memory Requested), and a table of server entries. The table columns are Name, Type, Health, Update Status, User, Registry, and Created. The table lists various servers such as Gmail, AntV Charts, santestnewmulti, Google Drive, and Nicks Test Compc.

**MCP Servers** [Sync] [+ Add MCP Server] [C]

Search servers...

Server Entries | Deployments & Connections | Registry Sources

**MCP Requested Resources**

Active Deployments: **76** | CPU Requested: **0** | Memory Requested: **14.8Gi**

<input type="checkbox"/>	Name	Type	Health	Update Status	User	Registry	Created	
<input type="checkbox"/>	Gmail	Remote	Available	Up to date		Global Registry	3 hours ago	...
<input type="checkbox"/>	Gmail	Remote	Available	Up to date		Global Registry	4 hours ago	...
<input type="checkbox"/>	AntV Charts	Single User	Available	Up to date		Global Registry	4 hours ago	...
<input type="checkbox"/>	santestnewmulti	Multi-User		Up to date		Global Registry	8 hours ago	...
<input type="checkbox"/>	AntV Charts	Single User	Available	Up to date		Global Registry	8 hours ago	...
<input type="checkbox"/>	Google Drive (Nicks Test Composi	Remote	Available	Up to date		Global Registry	9 hours ago	...
<input type="checkbox"/>	Nicks Test Compc	Composite		Up to date		Global Registry	9 hours ago	...
<input type="checkbox"/>	Atlassian	Remote	Available	Up to date		Global Registry	9 hours ago	...
<input type="checkbox"/>	Google Drive (Nicks Test Composi	Remote	Available	Up to date		Global Registry	9 hours ago	...
<input type="checkbox"/>	Nicks Test Compc	Composite		Up to date		Global Registry	9 hours ago	...



# Obot Gateway

MCP Server Curation & Creation

Deep Operational Visibility

**MCP Registry Access Control**

Audit Logs and Usage Statistics

Runtime Filters and Hooks

Infrastructure-level security

User Management and Role Based Access Control

The screenshot displays the 'MCP Registries' management page in the Obot Gateway. The interface is divided into a left sidebar and a main content area. The sidebar contains a navigation menu with the following sections: MCP Management (MCP Servers, MCP Registries, Audit Logs, Usage, Filters, Server Scheduling), Obot Agent Management (Token Usage, Model Providers, Model Access Policies, Skills, Skill Access Policies, Launch Agent), User Management (Users, Groups, User Roles, Auth Providers, API Keys), and Branding. The main content area is titled 'MCP Registries' and features a '+ Add New Registry' button. It is split into two sections: 'Admin Managed Registries' and 'User Managed Registries'. The 'Admin Managed Registries' table lists various registries with their server counts and delete icons. The 'User Managed Registries' table lists 'Default Access Rule' entries with no servers and no owners, each with a delete icon.

Name	Servers	Owner
Business Unit 1	11	
Business Unit 2	99	
Engineering Team 1	99	
Engineering Team 2	99	
Everyone	99	
IT Group 1	8	

Name	Servers	Owner
Default Access Rule	-	
Default Access Rule	-	
Default Access Rule	-	
Default Access Rule	-	
Default Access Rule	-	
Default Access Rule	-	
Default Access Rule	-	

# Obot Gateway

MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

**Audit Logs and Usage Statistics**

Runtime Filters and Hooks

Infrastructure-level security

User Management and Role Based Access Control

The screenshot shows the 'Audit Logs' page in the Obot Gateway. The sidebar on the left contains the following menu items:

- MCP Management
  - MCP Servers
  - MCP Registries
  - Audit Logs**
  - Usage
  - Filters
  - Server Scheduling
- Obot Agent Management
  - Token Usage
  - Model Providers
  - Model Access Policies
  - Skills
  - Skill Access Policies
  - Message Policies
  - Message Policy Violations
  - Agents
  - Launch Agent
- User Management
  - Users
  - Groups
  - User Roles
  - Auth Providers

The main content area is titled 'Audit Logs' and includes a search bar, a date range filter (3/25/26, 12:00 AM - 3/26/26, 11:59 PM), and buttons for '+ Create Export' and 'Manage Exports'. A filter for 'Call Type: tools/call OR prompts/get' is applied. Below the filters is a 'Timeline' bar chart showing usage over time. The chart shows a peak in activity on Wednesday, March 26, 2026, around 08:00 AM. Below the chart, there are 206 results on 1 page. The table below shows the first five entries:

#	TIMESTAMP	USER	SERVER	TYPE	IDENTIFIER	RES
1	Mar 26 2026 08:11:26 PM MST		nbalvqs89	tools/call	list_chats	20
2	Mar 26 2026 08:11:21 PM MST		nbalvqs89	tools/call	list_chats	20
3	Mar 26 2026 08:11:21 PM MST		nbalvqs89	tools/call	chat-with-nano...	20
4	Mar 26 2026 08:11:20 PM MST		nbalvqs89	tools/call	list_agents	20
5	Mar 26 2026 08:11:10 PM MST		Obot MCP Server	tools/call	obot_list_conne...	20

# Obot Gateway

MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

Audit Logs and Usage Statistics

**Runtime Filters and Hooks**

Infrastructure-level security

User Management and Role Based Access Control

The screenshot displays the 'Filters' management page in the Obot Gateway. The interface includes a sidebar with navigation options: MCP Management (MCP Servers, MCP Registries, Audit Logs, Usage, Filters, Server Scheduling), Obot Agent Management (Token Usage, Model Providers, Model Access Policies, Skills, Skill Access Policies, Launch Agent), User Management (Users, Groups, User Roles, Auth Providers, API Keys), and Branding. The main content area shows a table of filters with columns for Name, Webhook URL, and Selectors. Two filters are listed: 'Hate Speech Filter' and 'PII - Demo Filter (Source: https://github.com/cloudnautique/pii-demo-filter-server)'. A search bar and an '+ Add New Filter' button are located at the top right of the main area.

Name	Webhook URL	Selectors	
Hate Speech Filter		-	🗑️
PII - Demo Filter (Source: https://github.com/cloudnautique/pii-demo-filter-server)		1 selector	🗑️

# Obot Gateway

MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

Audit Logs and Usage Statistics

Runtime Filters and Hooks

**Infrastructure-level security**

User Management and Role Based Access

Control

The screenshot shows the 'Server Scheduling' configuration page in the Obot Gateway interface. The page is divided into three main sections: 'Tolerations Configuration', 'Resource Limits & Requests', and 'Runtime Class'. The left sidebar contains a navigation menu with categories like 'MCP Management', 'Obot Agent Management', 'User Management', and 'Branding'. The 'Server Scheduling' section is currently selected and highlighted.

**Server Scheduling**

Define the tolerations field for the pods in every MCP deployment. This value will be used to set the `spec.template.spec.tolerations` field on Kubernetes deployments and must be a valid list of [Toleration objects](#). See the Kubernetes [taints and tolerations documentation](#) for more details.

Tolerations Configuration

1

**Resource Limits & Requests** Helm-Deployed

Define the CPU and memory requests and limits for pods in every MCP deployment. See the Kubernetes [resource management documentation](#) for more information.

**CPU Settings**

Request: example: 500m      Limit: example: 1

**Memory Settings**

Request: example: 512Mi      Limit: example: 1Gi

**Runtime Class** Helm-Deployed

Specify a [RuntimeClass](#) for MCP server pods. RuntimeClass allows you to select a specific container runtime configuration for enhanced security isolation. Container runtimes like [gvisor](#) or [Kata Containers](#) provide stronger isolation by adding an additional security boundary between the container and the host kernel.

RuntimeClass Name: gvisor

Leave empty to use the cluster's default container runtime.

# Obot Gateway

MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

Audit Logs and Usage Statistics

Runtime Filters and Hooks

Infrastructure-level security

**User Management and Role Based Access**

**Control**

The screenshot shows the 'Users' management page in the Obot Gateway. The sidebar on the left contains the following menu items:

- MCP Management
  - MCP Servers
  - MCP Registries
  - Audit Logs
  - Usage
  - Filters
  - Server Scheduling
- Obot Agent Management
  - Token Usage
  - Model Providers
  - Model Access Policies
  - Skills
  - Skill Access Policies
  - Launch Agent
- User Management
  - Users (selected)
  - Groups
  - User Roles
  - Auth Providers
  - API Keys
- Branding

The main content area displays a table of users with the following columns: Name, Email, Assigned Role, Actual Role, Last Active, and Created. The table contains 14 rows of user data.

Name	Email	Assigned Role	Actual Role	Last Active	Created
Ar		Basic User	Basic User	17 days ago	18 days ago
		Basic User	Basic User	19 days ago	18 days ago
		Owner	Owner	5 days ago	27 days ago
		Basic User	Basic User	1 month ago	1 month ago
		Basic User	Basic User	4 days ago	1 month ago
		Basic User	Basic User	1 month ago	2 months ago
		Basic User	Basic User	6 days ago	2 months ago
		Owner	Owner	5 days ago	3 months ago
		Basic User	Basic User	4 months ago	4 months ago
		Basic User	Basic User	2 months ago	4 months ago
		Basic User	Basic User	6 months ago	6 months ago
		Basic User	Basic User	7 months ago	7 months ago
		Basic User	Basic User	7 months ago	7 months ago
C,		Basic User	Basic User	7 months ago	7 months ago

# Obot Gateway

MCP Server Curation & Creation

Deep Operational Visibility

MCP Registry Access Control

Audit Logs and Usage Statistics

Runtime Filters and Hooks

Infrastructure-level security

User Management and Role Based Access

Control

The screenshot displays the 'Users' management page in the Obot Gateway interface. A modal dialog titled 'Update Shannon Williams's Role' is open, allowing the user to select a new role. The background shows a table of users with columns for Name, Email, Assigned Role, Actual Role, Last Active, and Created. The left sidebar contains navigation menus for MCP Management, Obot Agent Management, User Management, and Branding.

**Update Shannon Williams's Role**

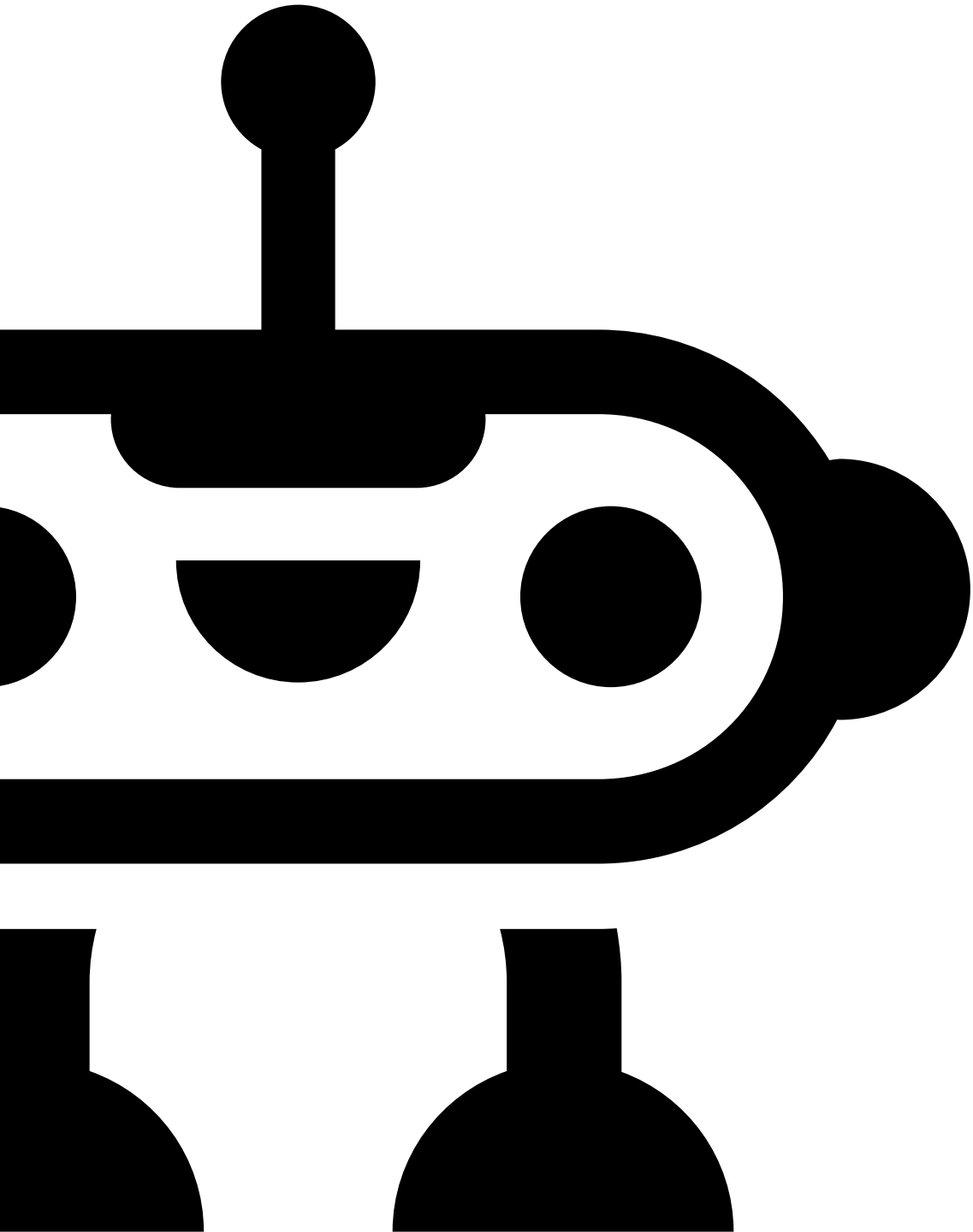
- Owner**
  - Owners can manage all aspects of the platform and can also assign the Owner role to other users.
- Admin**
  - Admins can manage all aspects of the platform.
- Power User+**
  - In addition to power user features, users can share their custom MCP servers through their own registries.
- Power User**
  - In addition to basic user features, users can publish custom MCP servers for their own personal use.
- Basic User**
  - Connect to MCP servers made available through registries and use Chat.
- Auditor**
  - Will have read-only access to the admin system and see additional details such as response, request, and header information in the audit logs.
- User Impersonation**
  - Will be able to connect to other users' Obot Agents. Requires Admin or Owner base role.

Buttons: Cancel, Update

Name	Email	Assigned Role	Actual Role	Last Active	Created
				7 months ago	7 months ago
				8 months ago	8 months ago
				4 months ago	8 months ago
				7 months ago	8 months ago
				Today	8 months ago
				1 day ago	8 months ago
				20 days ago	8 months ago
				7 days ago	8 months ago
				2 months ago	8 months ago
				7 months ago	8 months ago
				1 month ago	8 months ago
				3 months ago	8 months ago
		Owner, Auditor	Owner, Auditor	1 day ago	8 months ago
		Owner	Owner	1 month ago	8 months ago

# Citations and Resources

- <https://www.koi.ai/blog/postmark-mcp-npm-malicious-backdoor-email-theft>
- <https://www.wiz.io/blog/trivy-compromised-teampcp-supply-chain-attack>
- [https://www.trendmicro.com/en\\_us/research/26/c/inside-litellm-supply-chain-compromise.html](https://www.trendmicro.com/en_us/research/26/c/inside-litellm-supply-chain-compromise.html)
- <https://www.aquasec.com/blog/trivy-supply-chain-attack-what-you-need-to-know/>
- <https://github.com/aquasecurity/trivy/discussions/10425>
- <https://www.reversinglabs.com/blog/rl-identifies-malware-ml-model-hosted-on-hugging-face>
- <https://snyk.io/blog/poisoned-security-scanner-backdooring-litellm/>
- <https://snyk.io/blog/axios-npm-package-compromised-supply-chain-attack-delivers-cross-platform/>
- <https://github.com/nirholas/claude-code>
- [https://cheatsheetseries.owasp.org/cheatsheets/MCP\\_Security\\_Cheat\\_Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/MCP_Security_Cheat_Sheet.html)



**Thank You!**

<https://obot.ai>